arXiv:2405.00715v1 [cs.CL] 25 Apr 2024

Towards Adapting Open-Source Large Language Models for Expert-Level Clinical Note Generation

Hanyin Wang[*] **Chufan Gao** [†] **Bolun Liu**[*]

Qiping Xu [*] **Guleid Hussein**[*] **Mohamad El Labban** [‡] **Kingsley Iheasirim** [‡]

Hariprasad Korsapati [*] **Jimeng Sun** [†] [§]

[*]Division of Hospital Internal Medicine, Mayo Clinic Health System
[†]Department of Computer Science, University of Illinois Urbana-Champaign
[‡]Division of General Internal Medicine, Mayo Clinic Health System
[§]Carle Illinois College of Medicine, University of Illinois Urbana-Champaign
{wang.hanyin}@mayo.edu

Abstract

Large Language Models (LLMs) have shown promising capabilities in handling clinical text summarization tasks. In this study, we demonstrate that a small open-source LLM can be effectively trained to generate high-quality clinical notes from outpatient patient-doctor dialogues. We achieve this through a comprehensive domain- and task-specific adaptation process for the LLaMA-2 13 billion parameter model. This process incorporates continued pre-training, supervised fine-tuning, and reinforcement learning from both AI and human feedback. We introduced an enhanced approach, termed *DistillDirect*, for performing on-policy reinforcement learning with Gemini Pro serving as the teacher model. Our resulting model, LLaMA-Clinic, is capable of generating clinical notes that are comparable in quality to those authored by physicians. In a blinded physician reader study, the majority (90.4%) of individual evaluations rated the notes generated by LLaMA-Clinic as "acceptable" or higher across all three criteria: real-world readiness, completeness, and accuracy. Notably, in the more challenging "Assessment and Plan" section, LLaMA-Clinic scored higher (4.2/5) in real-world readiness compared to physician-authored notes (4.1/5). Additionally, we identified caveats in public clinical note datasets, such as ACI-BENCH. We highlight key considerations for future clinical note-generation tasks, emphasizing the importance of pre-defining a best-practice note format. Overall, our research demonstrates the potential and feasibility of training smaller, open-source LLMs to assist with clinical documentation, capitalizing on healthcare institutions' access to patient records and domain expertise. We have made our newly created synthetic clinic dialogue-note dataset and the physician feedback dataset publicly available to foster future research in this field.

1 Introduction

Recent breakthroughs in LLMs have transformed the landscape of natural language processing (NLP). Yet, we are only beginning to explore the use of LLMs in the medical field [1, 2]. Proprietary LLMs, such as GPT-4 and Med-PaLM, have demonstrated impressive capabilities in medical knowledge and

Preprint. Under review.

clinical NLP tasks [3–5]. However, most proprietary LLMs to date have limited flexibility for domain-specific fine-tuning, primarily due to restricted access to model weights. Additionally, proprietary LLMs raise several concerns pertinent to the healthcare sector, including HIPAA compliance, data security, cost, and transparency of training data [6–10].

The emergence of powerful open-source LLMs has opened up opportunities for domain-specific fine-tuning within the clinical field, yielding promising results [11–16]. For example, Meditron, a LLaMA-2 model pretrained on a vast corpus of curated medical literature, outperformed GPT-3.5 in the MedQA benchmark [16]. However, most research to date on open-source models has concentrated on medical knowledge injection rather than practical applications in real-world clinical workflow.

In this work, we address a practical question faced by clinicians in their everyday routine: *How can we best adapt an open-source LLM for the specific use case of clinical note generation?* Clinical note documentation represents a significant burden for healthcare practitioners [17] and appears to be a natural application for LLMs, given their remarkable generative capabilities [18]. Recent research on LLMs for clinical text summarization found that LLM-generated outputs are preferred over human summaries for their completeness and accuracy [5].

The 2023 ACL ClinicalNLP and CLEFImage workshops explored the generation of clinical notes from patient-doctor conversations using the newly released Ambient Clinical Intelligence Benchmark (ACI-BENCH) dataset [19, 20]. The most notable results were achieved using GPT-4 along with few-shot in-context learning [21, 5]. However, these initial exploratory studies revealed significant limitations: (1) The evaluation relied solely on automatic metrics of lexical similarity to the "reference" notes, such as ROUGE scores, which may not accurately reflect human preferences [22, 5]. (2) There was considerable variation in the quality, format, and style of "reference notes" within ACI-BENCH, suggesting that a note similar to an ACI-BENCH reference might not necessarily be of high quality for real-world application. (3) Previous studies conducted only limited SFT of open-source LLMs using small training datasets, thereby not fully exploring their potential for domain- and task-specific adaptation [19, 20, 5]. Additionally, the role of data augmentation and reinforcement learning remains unexplored.

In this study, we revisit the task of outpatient note generation and hypothesize that, through optimal training strategies, an open-source LLM can achieve performance in clinical note generation that is comparable to that of physicians. We thoroughly evaluated techniques for domain- and task-specific adaptation, ranging from continued pretraining and supervised fine-tuning (SFT) to reinforcement learning from AI and human feedback. We explored practical yet critical considerations for model training and deployment, drawing on our physician authors' expertise (such as the importance of defining a "best practice" note format). Lastly, we conducted an expert evaluation of our model's outputs against those from state-of-the-art proprietary models and physician-generated notes. Our resulting model, LLaMA-Clinic, demonstrated the capacity to generate clinical notes that are comparable in quality to those authored by physicians. The majority (90.4%) of individual evaluations by reviewers rated the notes generated by LLaMA-Clinic as "acceptable" or higher across all three criteria: real-world readiness, completeness, and accuracy. Notably, in the more challenging "Assessment and Plan" section, LLaMA-Clinic scored higher (4.2/5) in real-world readiness compared to physician-authored notes (4.1/5).

2 Results

2.1 Problem Formation

Given a recorded dialogue from a patient-doctor clinic encounter, we task LLMs to generate a high-quality outpatient note akin to one written by a clinician. This scenario is becoming increasingly prevalent due to the rising popularity of both in-person and virtual scribes, as well as automatic speech recognition devices in real-world settings [23]. Our research focuses solely on the generation of the "Subjective" and "Assessment and Plan" sections. This decision is informed by feedback from physician authors, who indicate that it would be unnatural or impractical during an actual clinical encounter to discuss all necessary information for the "Objective" part of a note, such as physical exam results. Moreover, much of the objective data, including lab and imaging results, are nowadays directly integrated into Electronic Medical Records (EMR). For each dialogue, two separate prompts are sent to the LLM to generate the "Subjective" and "Assessment and Plan" sections, respectively.

2.2 Experimental Design

Model Selection: We selected Meta's LLaMA-2 model, with 13 billion parameters (LLaMA-2-13B), as our foundational model for training [24]. We conducted experiments using both the base and chat models of LLaMA-2. Gemini Pro by Google was chosen as the proprietary model for generating reference notes and acting as the teacher model [25].

Experiments Overview: We demonstrated the experiments pipeline in Figure 1. Initially, we undertook domain-specific adaptation of LLaMA-2-13B through continued pretraining on discharge summaries from MIMIC-IV. This was followed by task-specific fine-tuning with SFT through instruction tuning. Next, we conducted reinforcement learning from AI feedback (RLAIF) using DistillDirect, our enhanced approach to performing direct preference optimization (DPO) on a distilled dataset. We then selected the model that performed best after SFT and RLAIF, as evaluated by the ROUGE-1 score against reference notes, for reinforcement learning from human feedback (RLHF) through DPO. Finally, notes generated by our ultimate model, denoted as LLaMA-Clinic, were reviewed by a panel of physicians against notes authored by physicians and Gemini Pro, in a blinded evaluation. All training was performed using Low-Rank Adaptation (LoRA). Detailed experimental methodologies are provided in the Methods section. Additionally, we have chronicled our complete experimental process in Supplementary Method 1, offering readers a comprehensive view of this complex trial-and-error journey.

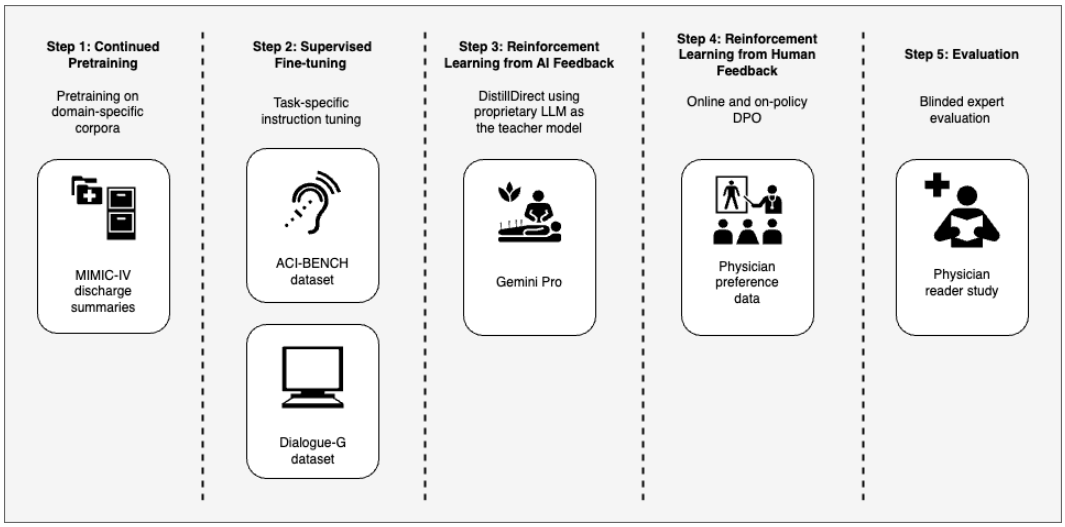


Figure 1: Overview of study design. We conducted a comprehensive domain- and task-specific adaptation process for the LLaMA-2-13B model. This process included continued pre-training, supervised fine-tuning, and reinforcement learning from AI and human feedback. Finally, we evaluated our model's outputs against those authored by physicians and Gemini Pro through a blinded expert evaluation. DPO: direct preference optimization.

2.3 Dataset and Preprocessing

Dataset: We used three datasets for our experiments.

- **ACI-BENCH:** ACI-BENCH represents the largest clinic dialogue-note dataset publicly available to date [23]. The dataset's dialogues were crafted by a team with medical expertise, and its clinical notes were initially generated using an automatic note generation system, then reviewed and revised by domain experts [23]. An observation from our study is the notable variation in the format, style, and quality of the "reference notes" within ACI-BENCH, especially in the section of "Assessment and Plan". While this diversity mirrors the reality of clinical practice, where different doctors may produce vastly different notes, it presents a challenge in using these notes as a "gold standard" for training an LLM to replicate. Consequently, we established a simple yet specific note format, recognized as "best practice" by a panel of licensed internal medicine physicians, to standardize our training approach. For model training, we retained only the dialogue section from ACI-BENCH

and employed Gemini Pro to generate notes based on the "best practice" format, serving as our reference notes. We demonstrated two examples of clinical notes before and after the change in Figure 2. The prompts to Gemini Pro for note generation are provided in Supplementary Method 3.

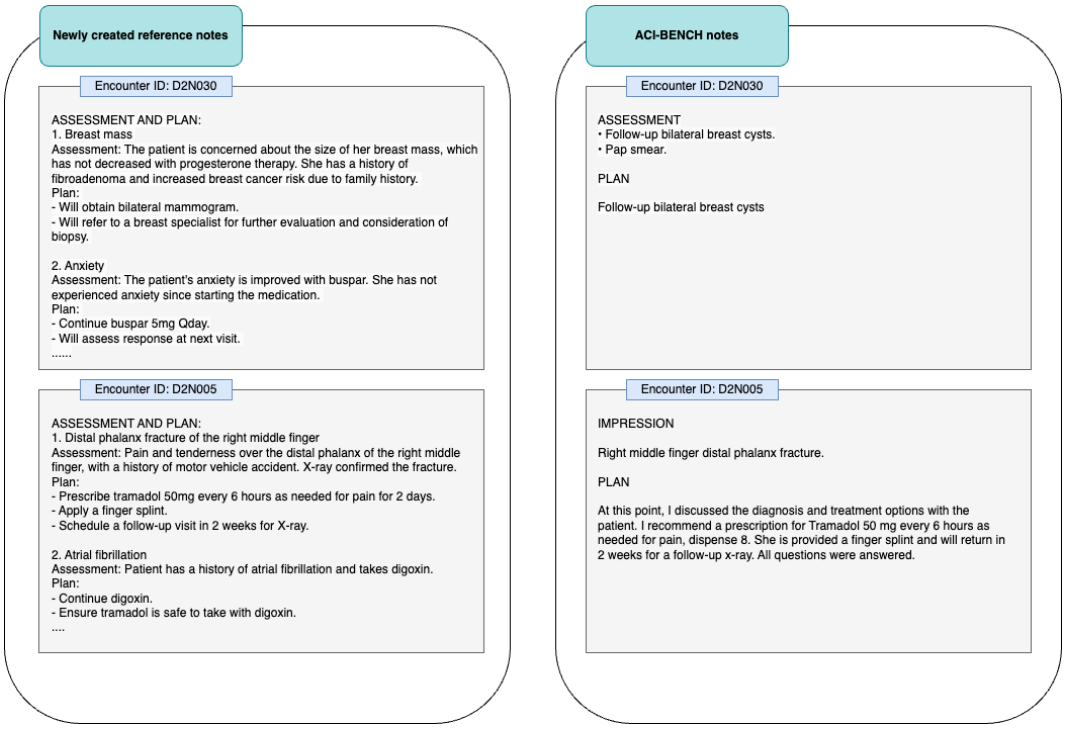


Figure 2: Comparison of newly created reference notes with ACI-BENCH notes. We compared newly generated reference notes using Gemini Pro with the original reference notes from two cases in ACI-BENCH. The newly created reference notes adhere more closely to our defined "best practice" format and contain more relevant medical information.

- **Dialogue-G:** In our efforts to enhance model training through data augmentation, we created a synthetic dataset of clinical dialogue-note pairs using Gemini Pro, which we have named Dialogue-G. We began by compiling a set of transcribed outpatient notes from the publicly available synthetic MTSamples dataset [26, 27] and utilized Gemini Pro to transform these notes into dialogues. Subsequently, we used these dialogues as inputs for Gemini Pro once again, this time to generate clinical notes based on the "best practice" format described above. We make the Dialogue-G dataset publicly available. The prompts to Gemini Pro for Dialogue-G creation are provided in Supplementary Method 2.
- **MIMIC-IV:** MIMIC-IV encompasses 431,231 unique hospital admissions from 299,712 patients who were admitted to either the Intensive Care Unit or the Emergency Department of Beth Israel Deaconess Medical Center in Boston, Massachusetts [28]. We utilized discharge summaries from MIMIC-IV for continued pretraining. Notably, the "brief hospital course" section of the discharge summaries is structurally akin to the "assessment and plan" section in outpatient notes. We compiled a subset of discharge summaries with only the "brief hospital course" using methods detailed in [29], referred to as "Discharge-short". We denoted the complete discharge summaries dataset as "Discharge-long".

Data Split:

- For continued pretraining, we explored both the Discharge-long dataset (1.2 billion tokens) and the Discharge-short dataset (0.2 billion tokens).
- We combined the training subsets from ACI-BENCH (dialogue n = 67) and Dialogue-G (dialogue n = 1291), then split this data equally for SFT and RLAIF, stratified by the data source. Note for each dialogue, there will be two separate reference notes for "Subject" and "Assessment and Plan", respectively.

- For RLHF, we utilized dialogues from the training, task3, and taskC subsets of ACI-BENCH (dialogue n = 147) as input prompts, and sampled from our model's outputs for human preference labeling. We excluded data from Dialogue-G to ensure in-distribution training at this final stage of model development, given the observed perplexity of 2.79 for Dialogue-G compared to 5.62 from ACI-BENCH (as calculated using the LLaMA-2 chat model post-continued pretraining).
- We examined results on the validation subset of ACI-BENCH throughout experiments. The final physician reader study was performed using the ACI-BENCH Test B subset (dialogue n = 40).

2.4 Analysis of Continued Pretraining

We presented the training loss curve in Figure 3. Across all experiments, the training loss rapidly decreased after the initial few hundred steps, then leveled off, showing minimal improvement thereafter. The lowest training loss achieved with the Discharge-long dataset is approximately 0.9, whereas, with the Discharge-short dataset, it remained around 1.4. According to scaling laws, a lower training loss is generally expected to correlate with improved performance in downstream tasks [30, 31]. The trajectories of training loss were similar for both the chat and base models. When experimenting with various training strategies and hyperparameters, we frequently observed training loss spikes that were slow to recover, as shown in Supplementary Figure 1. This phenomenon is well-documented in LLM pretraining [32]. We proceeded with models that did not exhibit loss spikes for further studies (i.e. our SFT and RLAIF experiments).

2.5 Analysis of SFT and RLAIF

As the primary objective of SFT and RLAIF is to align the output of LLaMA-2 with that of Gemini Pro, we evaluated the ROUGE-1 score, a measure of lexical similarity, against reference notes created by Gemini Pro. We reported the ROUGE-1 scores post-SFT and RLAIF, alongside those from the unmodified LLaMA-2, in Table 1. As anticipated, the vanilla chat model outperformed the base model out-of-box. However, continued pretraining with out-of-distribution corpora compromised the chat model's capacity to follow instructions. SFT significantly enhanced performance, particularly for the chat model over the base model. The model's performance was further boosted by RLAIF using DistillDirect. Notably, our training with DistillDirect frequently encountered instability—a well-known challenge in reinforcement learning [33]. We have detailed our experiments, including ablation studies, to find a stable training setup in Supplementary Method 1.3.

When comparing models pretrained with Discharge-long versus Discharge-short, the latter consistently scored higher, despite a higher training loss in the pretraining phase. The chat model pretrained with Discharge-short emerged as the top performer, and we selected this model checkpoint for the final RLHF stage. Interestingly, the vanilla LLaMA-13B models, directly subjected to SFT and RLAIF without prior continued pretraining, exhibited strong performances, including the highest score for "Subjective".

Qualitative analysis for a specific case is presented in Figure 4, aligning with the above quantitative findings. Continued pretraining effectively adopted the style and peculiarities from discharge summaries but at the expense of diminished instruction-following ability and increased hallucinations. The quality of outputs significantly improved post-SFT but remained overly verbose, while RLAIF effectively refined outputs to adhere to the format of reference notes, assisting in reducing hallucinations.

Upon manually reviewing outputs from all model checkpoints post-RLAIF, our physician author noted that the "Subjective" sections were generally of high quality and nearly indistinguishable. However, there is room for improvement in the "Assessment and Plan" sections to more accurately and concisely reflect medical reasoning.

2.6 Analysis of RLHF

We gathered quantitative feedback from three physician reviewers, who noted that the most frequent issues in the initial round of notes were inaccurate information or hallucinations. Two out of three reviewers observed that the quality of notes significantly improved with fewer hallucinations after one round DPO, describing notes as "often indistinguishable." Consequently, we conducted only two rounds of DPO, mindful of its time-intensive nature. The example in Figure 4 illustrates that RLHF

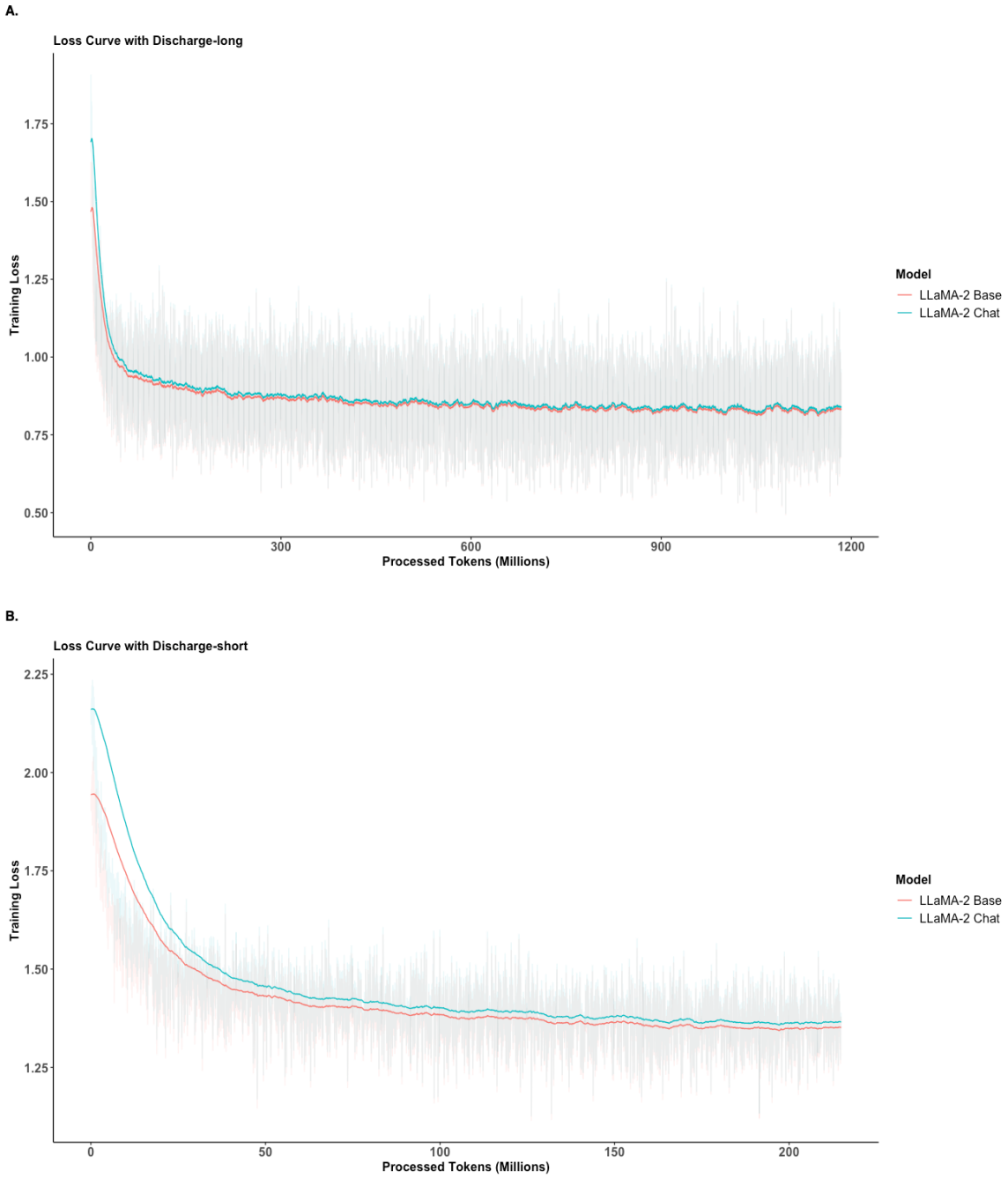


Figure 3: Training loss curve from continued pretraining. A. Training with the Discharge-long dataset (1.2 billion tokens). B. Training with the Discharge-short dataset (0.2 billion tokens). The X-axis represents processed training tokens, and the Y-axis represents training loss. The figures illustrate results from mixed precision training with a cosine learning rate scheduler. All experiments were trained for 1 epoch on their respective training datasets. The loss curve in the solid line was smoothed with an exponential moving average and a window size of 250 steps. The original loss values are shown as the faded background.

resulted in more granular changes, building on the results of RLAIF while maintaining the same output structures. We named the model after RLHF as LLaMA-Clinic for the physician reader study.

	Subjective			Assessment and Plan		
Models	CP	SFT	DistillDirect	CP	SFT	DistillDirect
Models underwent continued pretraining, SFT and RLAIF						
13B_short	0.1488	0.2463	0.4775	0.0899	0.2277	0.4956
13B_long	0.1612	0.2525	0.4494	0.1289	0.2565	0.4578
13B_chat_short	0.1520	0.3475	0.4878	0.1162	0.3055	**0.5182**
13B_chat_long	0.1708	0.3463	0.4601	0.1536	0.3620	0.4662
Models underwent SFT and RLAIF without continued pretraining						
13B	-	0.2813	**0.4994**	-	0.2977	0.4941
13B_chat	-	0.2874	0.4783	-	0.3301	0.4893
Out-of-box models*						
13B	0.0329			0.0100		
13B_chat	0.3585			0.4543		

Table 1: ROUGE-1 scores after SFT and RLAIF. All tests performed on validation subset of ACI-BENCH. Models including "chat" represent LLaMA-2 chat models, and otherwise represent LLaMA-2 base models. Models ended in "long" were pretrained using Discharge-long dataset, while models ended in "short" were pretrained using Discharge-short dataset. **Bolded scores** denote the best performance. *: Out-of-box models did not undergo CP, SFT or RLAIF. CP: continued pretraining. RLAIF: reinforcement learning from AI feedback. SFT: supervised fine-tuning.

2.7 Analysis of Physician Reader Study

We presented the results from the physician reader study in Figure 5. Two internal medicine physicians and one family medicine physician, in a blinded review, evaluated notes authored by physicians, LLaMA-Clinic, and Gemini Pro based on three criteria: real-world readiness, completeness, and accuracy (Figure 5). The median word counts and interquartile ranges (IQR) for notes authored by physicians, LLaMA-Clinic, and Gemini Pro were 118 (IQR: 94-150), 128 (IQR: 108-145), and 128 (IQR: 100-164), respectively. No statistically significant differences were observed among the three groups (Kruskal-Wallis H test: $p = 0.292$). We assessed inter-rater reliability (IRR) utilizing Gwet's AC2 statistics. The AC2 scores for the three metrics ranged from 0.80 to 0.83, signifying a high degree of agreement among reviewers. While minor score variances likely are not practically significant, Gemini Pro achieved the highest scores across all three criteria (Figure 5B). Remarkably, the majority (90.4%) of the individual evaluations rated the notes generated by LLaMA-Clinic as "acceptable" or higher across all three criteria (Figure 5C). Furthermore, the overall distribution of scores was also similar among the three groups (Figure 5C). This aligns with qualitative feedback from two out of three reviewers stating that the quality of notes was, for the "majority" of the time, indistinguishable among the groups.

The metric of real-world readiness yielded intriguing observations. Physicians were asked to evaluate the notes as though they were to be utilized in everyday clinical practice, under the assumption that they would proofread and make necessary edits. This inquiry was designed to assess the readiness of AI-generated notes in a practical workflow that places clinicians at the center and in full control. When compared to notes authored by physicians, LLaMA-Clinic received higher overall readiness scores, especially in the more challenging "Assessment and Plan" section, even though it scored slightly lower in accuracy (Figure 5b). This finding may suggest that in a real-world setting, physicians might be more tolerant of minor factual inaccuracies if the notes overall require fewer edits.

To assess the potential consequences of factual errors, we employed a blinded review process. One physician evaluated the likelihood and severity of harm associated with notes that received an accuracy or completeness score below 5 in his review (n = 22, 31 and 39 for Gemini Pro, LLaMA-Clinic, and Physician, respectively). All notes within the LLaMA-clinic and physician groups were assigned scores of "None" for the extent of harm and "Low" for the likelihood of harm. In contrast, the Gemini Pro group contained a single case rated with "Mild to Moderate" extent of harm and a "Medium" likelihood of harm.

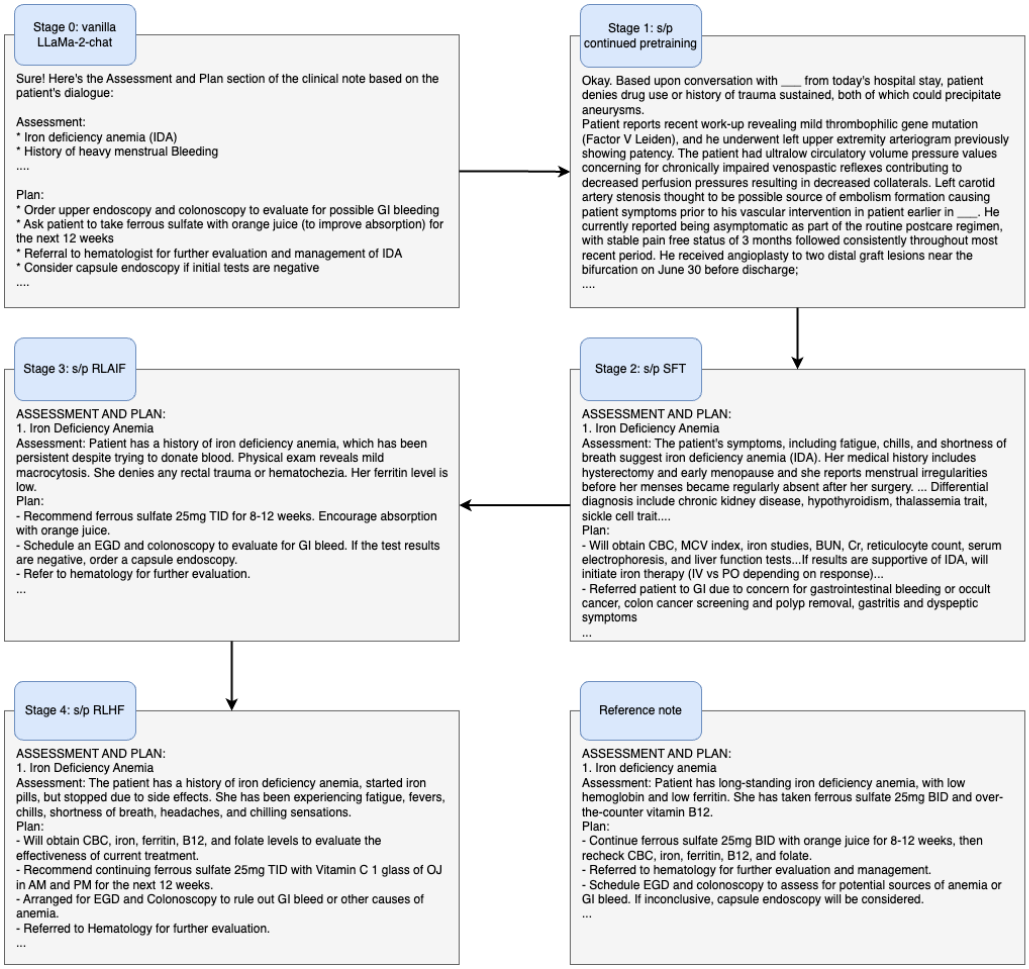


Figure 4: An example of model outputs progression with training steps. Case number D2N073 from ACI-BENCH validation set.

3 Discussion

Potential of Training LLMs for the Medical Domain: Our study highlights the potential and feasibility of training a smaller open-source LLM for the important task of generating clinical notes. With domain- and task-specific training steps, we have shown that even with a relatively small dataset and limited computing resources, it is possible to enhance an open-source LLM to perform comparably to physicians in this task. Distinguishing from prior research [19, 20, 5], our work differs in several aspects including the choice of model, training dataset, training strategies, and a focused alignment with real-world physician preferences.

Open Source vs. Proprietary Models: Adapting an open-source LLM offers several benefits. The accessibility to model weights lays the groundwork for flexible domain adaptation, including continued pretraining, SFT, and reinforcement learning exercised in this study. Such models can be hosted within healthcare institutions' firewalls, allowing for full control over model weights and mitigating concerns around data leakage and HIPAA violations. Furthermore, at inference times, models with smaller weights could offer faster and potentially more cost-effective performance with the same computing infrastructure [8, 9].

Considerations for the Note-Generation Task: Our methodology to adapt LLM for clinical note generation revealed several key considerations. We initiated this process by establishing a "best practice" note format, informed by a consensus among our physician authors. This approach tackles the issue of variability in the style, format, and quality of physician notes, which could otherwise compromise the training of LLMs. Instead of relying on the LLM to identify the most effective note

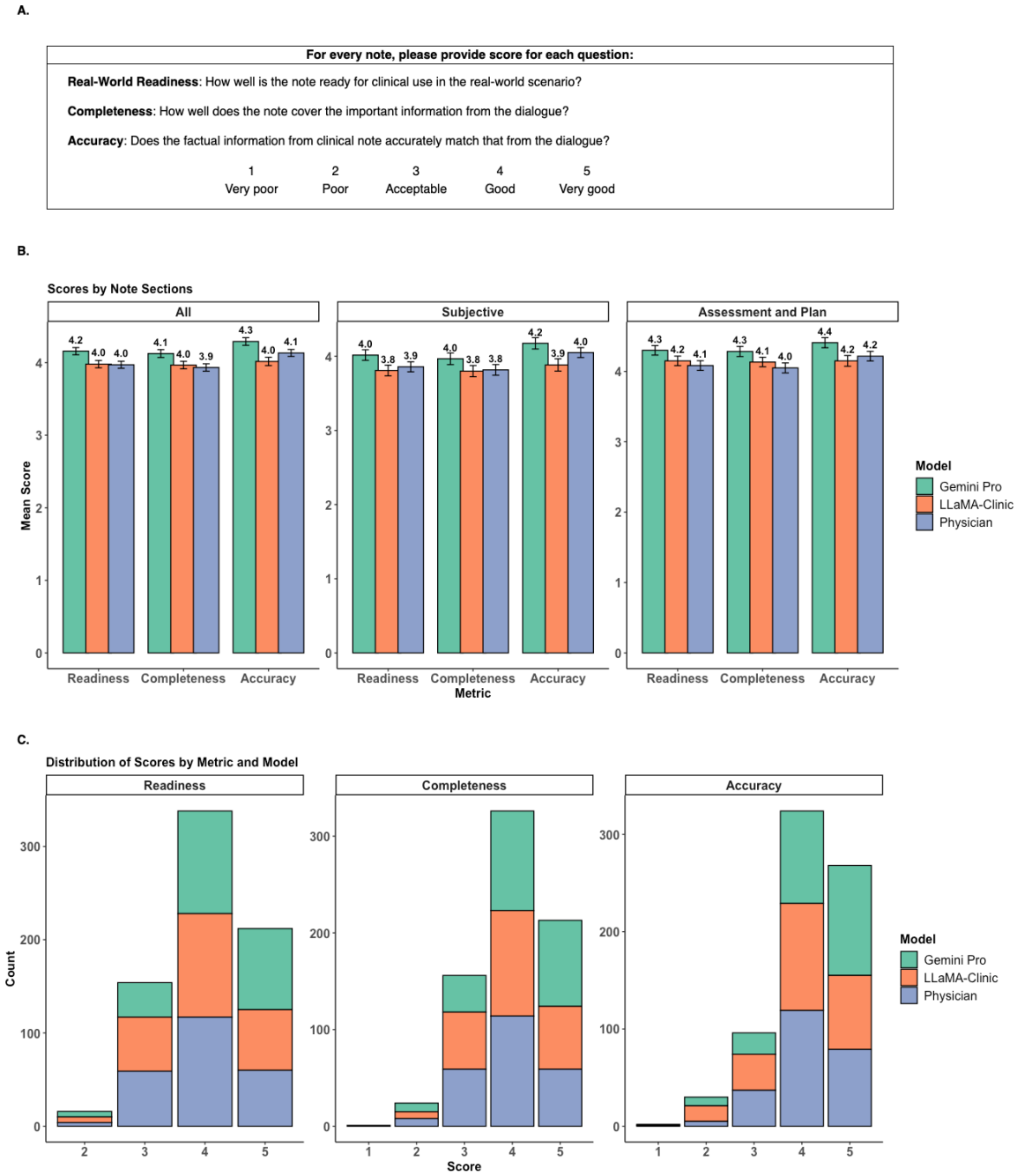


Figure 5: Physician Reader Study. A. Survey questions assessing each clinical note on three criteria: real-world readiness, completeness, and accuracy. B. Bar chart displaying the mean score among evaluators across different sections of the note. The error bars represent standard errors. The X-axis represents various metrics, and the Y-axis shows the mean scores. The subfigure labeled "All" displays the combined results. C. Distribution of scores across all criteria. The X-axis represents different scores, and the Y-axis shows total counts.

structure, we advocate for healthcare providers and institutions to set these standards themselves. It is important to acknowledge that the "best practice" note format employed in this study reflects the consensus of a group of internal medicine physicians. This format may not be generalizable to other specialties, such as orthopedic surgery. Future work that explores specialty-specific best practices for clinical note documentation could be highly beneficial. Furthermore, we recommend a workflow in which providers critically review and refine AI-generated notes prior to their filing. Under such a workflow, physician preferences may shift toward notes that generally require fewer revisions and

edits, with a higher tolerance for minor inaccuracies. This strategy emphasizes the critical role of physicians in meticulously evaluating all AI-generated content, thereby maintaining accountability for the clinical records.

Exploration of Best Practices to Adapt Open-souce LLM: The process of training a domain- and task-specific open-source LLM is inherently complex, involving numerous engineering decisions without a clear consensus on best practices. These uncertainties span a range of decisions, including, but not limited to, selecting from various open-source models (e.g., LLaMA-2 [24], Mixtral [34], Gemma [35]), choosing between base and chat models, determining the most effective training corpora, selecting data for SFT and reinforcement learning, and exploring extensive combinations of hyperparameters. We have documented our complete experimental journey in Supplementary Method 1 to provide readers with insight into this trial-and-error process. Below, we discuss several key technical considerations encountered during model training.

Continued Pretraining and SFT: Continued pretraining of an LLM using a domain-specific corpus is recognized for enhancing performance on downstream tasks [36, 37]. This phase is considered a knowledge injection process, given that LLMs acquire the vast majority of their knowledge during the pre-training phase [38, 39]. Several clinical LLMs that underwent continued pretraining with medical corpora, such as PubMed literature, combined with SFT have shown significant improvements in medical knowledge benchmarks [16, 14, 40].

However, a critical distinction exists between tasks focused on medical knowledge (e.g., answering USMLE questions) and those aimed at clinical note generation. Notably, GatorTronGPT [41], the only LLM trained from scratch using EMR data from real patients to date, performed lower in both MedQA and PubMedQA compared to other clinical LLMs [16, 14, 37]. This outcome indicates that EMR data alone may lack comprehensive medical knowledge. For the task of note generation, we hypothesized that continued pretraining with clinical notes could offer benefits by introducing greater lexical variance, unique semantic patterns, and diverse formatting similar to prior work [42]. Our experiments did not conclusively demonstrate the anticipated benefits of continued pretraining, as the out-of-the-box LLaMA-2 model without continued pretraining achieved the highest ROUGE-1 scores after SFT and RLAIF (Supplementary Table 3). We opted to proceed with the continued pretrained model for RLHF due to subtle peculiarities observed upon manual inspection, though these differences were minor (Supplementary Method 1.3). Given the significant time and computational resources required for continued pretraining, its utility, particularly with clinical notes, merits further exploration in future work.

Data Selection for Continued Pretraining: Another potential factor in the less impressive improvement from continued pretraining may be attributed to the variance in data distribution between discharge summaries and outpatient notes. To address this, we performed experiments focused on a condensed version of discharge summaries, hypothesizing that the "brief hospital course" section would contain data of higher quality than the complete discharge summary. Indeed, models trained on the "brief hospital course" section outperformed those trained on the full summaries (Table 1). Interestingly, it was observed that during the pretraining stage, models trained on the full summaries achieved lower training loss (Figure 3), which, however, did not correspond to enhanced performance in the downstream task. We speculate that the structured nature of the full discharge summary, with its sections such as laboratory results, medications, and post-discharge instructions, presents more straightforward learning targets for the model, enabling it to achieve lower training losses but not necessarily translate to better task performance. This observation underscores the necessity of a thorough analysis of the data used for training.

RLAIF and RLHF: The most notable performance improvement was observed during the RLAIF and RLHF stages. The advent of DPO has considerably streamlined the process of reinforcement learning by obviating the need for training an explicit reward model. However, best practices for implementing DPO are yet to be established. Our approach, which stages reinforcement learning using DPO into RLAIF and RLHF, draws inspiration from recent advancements in distilled DPO, adversarial preference optimization, and online AI feedback [43–45]. It offers several enhancements. Firstly, for each iteration of RLAIF with DistillDirect, we used the model's outputs as "reject samples" to inform feedback, ensuring on-policy learning as opposed to the off-policy training in previous work [44]. Secondly, during the RLHF phase, we gathered human preferences on responses generated by the most recent model's outputs, promoting online and on-policy training. With carefully selected training hyperparameters, DistillDirect may achieve performance improvements without overfitting,

despite the limited training data and repetitive use of the same prompts in each training cycle (see Supplementary Table 4). Prior work suggested that RLHF has robust generalization capabilities, even when facing distribution shifts between training and test data, albeit at the expense of output diversity [46].

Limitations: Our study serves as a proof of concept and encounters limitations, notably the scarcity of publicly available patient-physician dialogues for model training, with the latest ACI-BENCH dataset comprising fewer than 300 cases. Additionally, the effectiveness of outpatient note generation is fundamentally linked to the content of patient-physician dialogues. Although the ACI-BENCH data may have synthesized dialogues to encapsulate all necessary information for composing a comprehensive note, such ideal conditions may not always reflect real-world scenarios. For instance, time constraints may prevent physicians from discussing every detail of medical reasoning with patients, potentially degrading the quality of the generated notes. Our research was also constrained by limited academic computing resources and the availability of physician evaluators, which restricted our ability to conduct extensive hyperparameter searches or additional rounds of RLHF. Our final evaluation was conducted with only three physicians due to the task's time-intensive nature. Ideally, future developments would involve experiments with the latest powerful open-source models (e.g., Mixtral [34]), continued pretraining on in-domain and in-distribution data (e.g., outpatient notes rather than inpatient discharge summaries), SFT and RLAIF on larger amount of real patient-doctor dialogues, RLHF until a performance plateau is observed, and evaluation by a diverse group of medical professionals.

Conclusions: Our research underscores the potential of training an open-source LLM for outpatient note generation with promising clinical applicability. The domain- and task-specific adaptation processes, such as continued pretraining, SFT, and reinforcement learning, are uniquely feasible with open-source models. Healthcare institutions are in a privileged position to undertake such endeavors, given their access to extensive EMR data and a wealth of domain expertise critical for the implementation of RLHF. Notably, our work was based on fewer than 1,500 patient-doctor dialogues and limited physician preference data. When implementing a similar project in a healthcare institution, these training data could reasonably be scaled up by several orders of magnitude, which would likely boost performance further. Furthermore, the prospect of conducting similar work for other clinical note-generation tasks, such as creating discharge summaries for hospitalized patients, is particularly exciting.

4 Methods

4.1 Low-Rank Adaptation (LoRA)

We used LoRA to train LLaMA-2-13B models for all phases of training. LoRA is a method that involves freezing the pre-trained model weights and only training a small percentage (<1%) of the model weights; i.e. by incorporating trainable rank decomposition matrices into each layer of the transformer architecture [47]. As a quick summary, let us assume that we have the original weight matrix $\boldsymbol{W_0} \in \mathbb{R}^{d\times k}$. LoRA works by adding a low-rank matrix to the original weight matrix: $\Delta\boldsymbol{W} + \boldsymbol{W_0}, \Delta\boldsymbol{W} = \boldsymbol{BA}$ where $\boldsymbol{B} \in \mathbb{R}^{d\times r}$ and $\boldsymbol{A} \in \mathbb{R}^{r\times k}$. $r << d$, so the matrices $\boldsymbol{B}, \boldsymbol{A}$ are limited by a lower rank r, reducing the need to train all the parameters. Training is only performed on this $\Delta\boldsymbol{W}$, and original model weights are kept the same.

Training Details: In all steps of training, LoRA parameters were configured with r set to 8, an alpha value of 32, and a dropout rate of 0.05. All attention blocks were included in the Lora target modules.

4.2 Continued Pretraining

Continued pretraining extends the initial pretraining process of an LLM, using the same autoregressive objective to maximize the likelihood of predicting the next token. For LLaMA-2-13B, this objective is next token prediction, based on all previously predicted tokens. Research from LIMA indicated that the majority of knowledge embedded in LLMs is acquired during the pretraining stage [39]. In the process of domain adaptation, it is a common practice to employ pretraining or continued pretraining using domain-specific corpora to improve performance in downstream tasks [16, 37]. In our work, we experimented with continued pretraining on the MIMIC-IV discharge summaries.

Training Details: We followed the training scripts outlined in Meta's official LLaMA recipe repository [48]. We employed mixed-precision training with a batching strategy of packing and a context length of 4096 tokens. We utilized Fully Sharded Data Parallel (FSDP) on either 4 Nvidia A6000 or 4 Nvidia A100 GPUs. We maintained a batch size of 4 during training with a gradient accumulation step of 1. Consistent with LLaMA-2, we set a peak learning rate of 3e-4 for the continued pre-training stage. The AdamW optimizer with a cosine learning rate scheduler was used, and the model was trained for one epoch. The exponential moving average of training loss as shown in Figure 3 was calculated using the `pracma` package from R with a window size of 250 [49].

4.3 Supervised Finetuning (SFT)

We implemented SFT in the format of instruction tuning, utilizing the ACI-BENCH (with newly created reference notes) and Dialogue-G datasets. Instruction tuning plays a crucial role in the task-specific adaptation of an LLM, facilitating the model's learning of specific tasks through guided instructions [50]. In this approach, each sample consists of a prompt and an answer, with our study employing patient-doctor dialogues as prompts and clinical notes as answers. SFT employed the same autoregressive objective as continued pretraining, but losses on tokens from the prompt were zeroed out, consistent with the approach used in LLaMA-2 [24].

Training Details: We used a similar experiment setup as continued pretraining, including following LLaMA-recipes to perform mixed precision training on 4 GPUs using FSDP. For SFT, we selected the batching strategy of padding and trained on 3 epochs. Consistent with LLaMA-2, we set a peak learning rate of 2e-5. We truncate prompt (including instruction and dialogue) at a max length of 3000 tokens, and truncate note to a max length of 1000 tokens. We set a value of -100 for labels on prompt tokens to zero out losses from prompts.

4.4 Reinforcement Learning with AI Feedback (RLAIF)

Here, we begin by reviewing related works on direct preference optimization and distilled direct preference optimization. Subsequently, we introduce our enhanced approach, termed DistillDirect, which is utilized in this study for RLAIF.

Direct Preference Optimization (DPO): Reinforcement Learning with Human Feedback (RLHF) has been instrumental in the success of LLMs like InstructGPT [51]. However, conventional RLHF strategies, such as proximal policy optimization (PPO), are marked by their technical complexity and the high demand for computational resources [52]. The emergence of direct alignment techniques, notably DPO, has greatly streamlined this process by obviating the need for training a reward model [53]. DPO begins by collecting a preference dataset $\mathbb{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i^+, \boldsymbol{y}_i^-)\}_{i=1}^N$, where for each prompt $\boldsymbol{x}$, there is a preferred answer $\boldsymbol{y}^+$ and a rejected answer $\boldsymbol{y}^-$. Following the notations as in [45], DPO optimizes the language model (target policy $\pi_{\boldsymbol{\theta}}$) using the following loss function:

$$-\log \sigma \left(\beta \log \frac{\pi_{\boldsymbol{\theta}}(\boldsymbol{y}^+|\boldsymbol{x})\pi_{\boldsymbol{\theta}^0}(\boldsymbol{y}^-|\boldsymbol{x})}{\pi_{\boldsymbol{\theta}^0}(\boldsymbol{y}^+|\boldsymbol{x})\pi_{\boldsymbol{\theta}}(\boldsymbol{y}^-|\boldsymbol{x})} \right) \tag{1}$$

Here, $\pi_{\boldsymbol{\theta}^0}$ represents the SFT baseline used as a reference, σ denotes the logistic function, and β is introduced as a scalar hyperparameter. Intuitively, this loss function increases the probability of the latest model $\pi_{\boldsymbol{\theta}}$ generating preferred response $\boldsymbol{y}^+$ over the original model $\pi_{\boldsymbol{\theta}^0}$. Similarly, it decreases the probability of the latest model $\pi_{\boldsymbol{\theta}}$ generating rejected response $\boldsymbol{y}^-$ over the original model $\pi_{\boldsymbol{\theta}^0}$.

Distilled Direct Preference Optimization (dDPO): The original dDPO methodology, applied in training Zepher, involves generating a collection of responses for each prompt from various LLMs [44]. These responses are then evaluated by a teacher model (GPT-4) to provide preference feedback. The application of RLAIF with dDPO has yielded positive outcomes in Zepher, notably enhancing intent alignment. However, dDPO's reliance on a pre-collected preference dataset renders it suboptimal due to its off-policy and offline characteristics, which are elaborated more in the next section.

Online vs. Offline Training and On-policy vs. Off-policy Training: These terms describe the nuances in the creation of a preference dataset $\mathbb{D}$. For any given prompt $\boldsymbol{x}$, initially, two responses $\boldsymbol{y}^1$ and $\boldsymbol{y}^2$ are generated from an LLM denoted as ρ. These responses are then assessed for preference by humans or AI, being labeled as $\boldsymbol{y}^+$ (preferred) and $\boldsymbol{y}^-$ (rejected). In this context, training is

considered *on-policy* if $\rho = \pi_{\theta}$, or when the generated responses are sampled from the latest version of the LLM during RLAIF training. *Off-policy* training indicates otherwise. Learning is deemed *online* if the preference labeling and training is conducted in real-time, directly on the outputs from the currently trained policy. It is considered *offline* if preference labeling and training are performed in separate, discrete steps.

Given the significant time and financial costs associated with collecting preference data, utilizing pre-collected preference datasets, such as those employed in dDPO, is a common practice. Defined by the criteria mentioned above, this method falls under the category of offline, off-policy learning. A critical caveat of employing a pre-collected preference dataset for offline and off-policy training is the occurrence of distribution shifts [45]. More specifically, distribution shifts arise at time step t because the preferred and rejected response is sampled from a policy ρ, where $\rho \neq \pi_{\theta^t}$. Research has shown that online RLAIF systems, which are designed to mitigate these distribution shifts, significantly surpass the effectiveness of offline DPO methodologies [45].

RLAIF using DistillDirect: In our study, we introduce an improved approach based on dDPO, termed DistillDirect, specifically designed to ensure on-policy learning on a distilled dataset. In each training cycle, we begin by sampling a response from π_{θ^t}, ensuring that the learning process remains strictly on-policy. This sampled response is then designated as the rejected response, while a reference response from Gemini Pro (the teacher model) is considered the preferred outcome. This approach implicitly assumes that the response from π_{θ^t} is generally less favored than that from the teacher model—an assumption that we validated through manual review in each round of training.

Training details: We utilized the `trl` library from Huggingface to conduct DistillDirect [54]. Due to computational limitations, experiments were conducted on a single Nvidia A100 GPU with 80GB of graphics memory. To optimize memory usage, pure BF16 training was utilized with a micro-batch size of 1 and gradient accumulation steps of 8. Following a limited learning rate search detailed in Supplementary Method 1.3 and Supplementary Table 4, a learning rate of 5e-6 was chosen. The optimizer used was `paged_adamw_32bit`. Within the `DPOTrainer` class, we set the beta hyperparameter to 0.1 and passed None to `ref_model`. Three rounds of DistillDirect were performed, with each round involving one epoch of training.

Text generation was implemented using the Transformers library [55]. When generating "rejected" samples from the latest model checkpoint, we consistently applied do_sample=True, top_p=1.0, temperature=1.0, top_k=50, repetition_penalty=1.2, and use_fast_kernels=False. The maximum number of newly generated tokens was set to 1000.

4.5 Reinforcement Learning from Human Feedback (RLHF)

In the concluding phase of our training process, we select the most effective model from the preceding stage for further online and on-policy learning using DPO.

Physician Preference Data Collection: In each round of DPO, for a specific prompt $\boldsymbol{x}$, three responses are generated from π_{θ^t} and evaluated by our physician reviewers. Three licensed internal medicine physicians are tasked with providing preference feedback by selecting the most and least preferred responses, with criteria focusing on clinical readiness, correctness, and adherence to the desired format. A notable adaptation in our approach is that reviewers are also instructed to make minor adjustments to improve the quality of the preferred responses, such as correcting factual inaccuracies. Detailed instructions for collecting preference data are available in Supplementary Method 4.

Training details: For RLHF, we employed an experimental setup analogous to that described in RLAIF. We conducted two rounds of DPO on human preference data. Diverging from the approach taken in RLAIF, we executed three epochs of training in each DPO round due to the limited size of the dataset.

Based on the findings from ablation studies detailed in Supplementary Method 1.3 and Supplementary Table 5, we opted for a lower temperature setting at this stage. In the initial round of DPO, we generated three responses using the same configuration, including a temperature setting of 0.6, for preference labeling. In the second round of DPO, we diversified the temperature settings, resulting in three responses with temperatures set at 0.6, 0.4, and 0.2, respectively, for preference labeling.

4.6 Physician Reader Study

The three internal medicine physicians engaged in preference data collection were tasked with writing clinical notes based on conversations from the ACI-BENCH TestB subset, adhering to the pre-defined "best practice" format. AI-generated notes were produced by LLaMA-Clinic and Gemini Pro, employing identical generation-related hyperparameters (temperature of 0.2, top_p of 0.7 and top_k of 40). To ensure consistency in presentation across all notes, we implemented a basic post-processing step. This step standardized aspects like style, spacing, and capitalization to minimize any formatting discrepancies between human-authored and model-generated notes.

The physician-authored notes, alongside those generated by Gemini Pro and LLaMA-Clinic, were reviewed by three additional physicians, who were not involved in the preference labeling. These licensed physicians, who specialize in general internal medicine or family medicine, boast rich experience in outpatient practice. The notes were presented in a random order, anonymized to remove any identifying information, and labeled as note 1, note 2, and note 3 to mask the origin of each note from the evaluators. Before assessing the notes, evaluators were instructed to read the entire patient-provider conversation. They were then asked to rate the quality of each note across three criteria: "accuracy," "completeness," and "real-world readiness." For each criterion, a scoring system from 1 to 5 was used, ranging from very poor to very good, with higher scores reflecting superior quality. Specifically for "real-world readiness," evaluators were prompted to consider the scenario of integrating AI-generated clinical notes into their daily practice, including the necessity to proofread and potentially edit these notes before filing them into medical records. Detailed instructions and the scoring rubric are available in the Supplementary Method 5. For the harm evaluation, we followed the same approach as in [5, 56] by asking one physician reviewer to assess all notes with an accuracy or completeness score of less than 5 and answer two questions related to the extent and likelihood of harm. The instruction for harm evaluation can be found in Supplementary Method 6.

4.7 Statistical Analysis

The non-parametric Kruskal-Wallis H test was selected to compare differences in word counts among the three-note groups, utilizing the `scipy` package in Python [57]. We measured IRR using Gwet's AC2 statistics implemented through the `irrCAC` package in R [58, 59]. We reported results with quadratic weights for Gwet's AC2, as this approach is reliable for ordinal data against the grey zones [60]. Due to our relatively small reviewer pool, we chose not to conduct statistical significance testing on the physician reader study, aligning with practices observed in the deep learning community [24, 61].

5 Data availability

Access to MIMIC-IV can be requested at `https://physionet.org/content/mimiciv/`, which requires a signed safe usage agreement. ACI-BENCH dataset can be accessed at `https://github.com/wyim/aci-bench`. Dialogue-G, physician preference data, and the final test data can be found at `https://github.com/hanyin88/llama-clinic`.

6 Code availability

Scripts for this work were written in Python. They are available with accompanied documentation at `https://github.com/hanyin88/llama-clinic`.

7 Acknowledgement

This research was supported by NSF awards SCH-2205289, IIS-2034479, and SCH-2014438. The funder played no role in the study design, data collection, analysis, and interpretation of data, or the writing of this manuscript. This research is also supported by the Mayo Clinic Health System Southwest Minnesota Region Protected Research Grant.

8 Competing interests

The authors declare no competing interests.

9 Author Contributions

H.W. designed, conducted, and analyzed the results of experiments. H.W. and C.G. wrote the original draft. J.S. obtained funding and computing resources for the project. L.B., Q.X., and H.G. provided human preference labeling for RLHF and wrote notes for the physician reader study. M.E., K.I., and H.K. conducted the physician reader study. H.K. also performed the harm evaluation. All authors contributed to the conceptualization of the research questions. All authors reviewed, revised, and approved the final manuscript.

References

[1] Kai He, Rui Mao, Qika Lin, Yucheng Ruan, Xiang Lan, Mengling Feng, and Erik Cambria. A survey of large language models for healthcare: from data, technology, and applications to accountability and ethics. *arXiv preprint arXiv:2310.05694*, 2023.

[2] Hongjian Zhou, Boyang Gu, Xinyu Zou, Yiru Li, Sam S Chen, Peilin Zhou, Junling Liu, Yining Hua, Chengfeng Mao, Xian Wu, et al. A survey of large language models in medicine: Progress, application, and challenge. *arXiv preprint arXiv:2311.05112*, 2023.

[3] Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*, 2023.

[4] Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, et al. Towards expert-level medical question answering with large language models. *arXiv preprint arXiv:2305.09617*, 2023.

[5] Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, William Collins, Neera Ahuja, et al. Clinical text summarization: Adapting large language models can outperform human experts. *arXiv preprint arXiv:2309.07430*, 2023.

[6] Mason Marks and Claudia E Haupt. Ai chatbots, health privacy, and challenges to hipaa compliance. *Jama*, 2023.

[7] Jan Clusmann, Fiona R Kolbinger, Hannah Sophie Muti, Zunamys I Carrero, Jan-Niklas Eckardt, Narmin Ghaffari Laleh, Chiara Maria Lavinia Löffler, Sophie-Caroline Schwarzkopf, Michaela Unger, Gregory P Veldhuizen, et al. The future landscape of large language models in medicine. *Communications medicine*, 3(1):141, 2023.

[8] Alaa Dania Adimi. Comparison & cost analysis: Should we invest in open-source or closed-source llms? `https://medium.com/@ja_adimi`, Jan 2024. Accessed: March, 2024.

[9] Ben Wodecki. Open-source vs closed models: The true cost of running ai. `https://aibusiness.com/nlp/open-source-vs-closed-models-the-true-cost-of-running-ai`, Nov 2023. Accessed: April, 2024.

[10] Ryan Mac Michael M. Grynbaum. The times sues openai and microsoft over a.i. use of copyrighted work. `https://www.nytimes.com/2023/12/27/business/media/new-york-times-open-ai-microsoft-lawsuit.html`, Dec 2023. Accessed: Jan, 2024.

[11] Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. Medalpaca–an open-source collection of medical conversational ai models and training data. *arXiv preprint arXiv:2304.08247*, 2023.

[12] Haochun Wang, Chi Liu, Nuwa Xi, Zewen Qiang, Sendong Zhao, Bing Qin, and Ting Liu. Huatuo: Tuning llama model with chinese medical knowledge. *arXiv preprint arXiv:2304.06975*, 2023.

[13] Li Yunxiang, Li Zihan, Zhang Kai, Dan Ruilong, and Zhang You. Chatdoctor: A medical chat model fine-tuned on llama model using medical domain knowledge. *arXiv preprint arXiv:2303.14070*, 2023.

[14] Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. Pmc-llama: Further finetuning llama on medical papers. *arXiv preprint arXiv:2304.14454*, 2023.

[15] Augustin Toma, Patrick R Lawler, Jimmy Ba, Rahul G Krishnan, Barry B Rubin, and Bo Wang. Clinical camel: An open-source expert-level medical language model with dialogue-based knowledge encoding. *arXiv preprint arXiv:2305.12031*, 2023.

[16] Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, et al. Meditron-70b: Scaling medical pretraining for large language models. *arXiv preprint arXiv:2311.16079*, 2023.

[17] Elske Ammenwerth and H-P Spötl. The time needed for clinical documentation versus direct patient care. *Methods of information in medicine*, 48(01):84–91, 2009.

[18] Peter Lee, Sebastien Bubeck, and Joseph Petro. Benefits, limits, and risks of gpt-4 as an ai chatbot for medicine. *New England Journal of Medicine*, 388(13):1233–1239, 2023.

[19] Asma Ben Abacha, Wen-wai Yim, Griffin Adams, Neal Snider, and Meliha Yetisgen-Yildiz. Overview of the mediqa-chat 2023 shared tasks on the summarization & generation of doctor-patient conversations. In *Proceedings of the 5th Clinical Natural Language Processing Workshop*, pages 503–513, 2023.

[20] Wen-wai Yim, A Ben Abacha, N Snider, G Adams, and Meliha Yetisgen. Overview of the mediqa-sum task at imageclef 2023: Summarization and classification of doctor-patient conversations. In *CLEF*, 2023.

[21] John Giorgi, Augustin Toma, Ronald Xie, Sondra Chen, Kevin An, Grace Zheng, and Bo Wang. Wanglab at mediqa-chat 2023: Clinical note generation from doctor-patient conversations using large language models. In *Proceedings of the 5th Clinical Natural Language Processing Workshop*, pages 323–334, 2023.

[22] Francesco Moramarco, Alex Papadopoulos Korfiatis, Mark Perera, Damir Juric, Jack Flann, Ehud Reiter, Anya Belz, and Aleksandar Savkov. Human evaluation and correlation with automatic metrics in consultation note generation. *arXiv preprint arXiv:2204.00447*, 2022.

[23] Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific Data*, 10(1):586, 2023.

[24] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[25] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[26] MTSamples. Transcribed medical transcription sample reports and examples. `https://mtsamples.com/`. Accessed: Nov, 2023.

[27] Yan Hu, Qingyu Chen, Jingcheng Du, Xueqing Peng, Vipina Kuttichi Keloth, Xu Zuo, Yujia Zhou, Zehan Li, Xiaoqian Jiang, Zhiyong Lu, et al. Improving large language models for clinical named entity recognition via prompt engineering. *Journal of the American Medical Informatics Association*, page ocad259, 2024.

[28] Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1, 2023.

[29] Hanyin Wang, Chufan Gao, Christopher Dantona, Bryan Hull, and Jimeng Sun. Drg-llama: tuning llama model to predict diagnosis-related group for hospitalized patients. *npj Digital Medicine*, 7(1):16, 2024.

[30] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[31] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[32] Stas Bekman. Machine Learning Engineering Online Book. `https://github.com/stas00/ml-engineering`, 2023. Accessed: Nov, 2023.

[33] Zihan Ding and Hao Dong. Challenges of reinforcement learning. *Deep Reinforcement Learning: Fundamentals, Research and Applications*, pages 249–272, 2020.

[34] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[35] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

[36] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*, 2023.

[37] Sunjun Kweon, Junu Kim, Jiyoun Kim, Sujeong Im, Eunbyeol Cho, Seongsu Bae, Jungwoo Oh, Gyubok Lee, Jong Hak Moon, Seng Chan You, et al. Publicly shareable clinical large language model built on synthetic clinical notes. *arXiv preprint arXiv:2309.00237*, 2023.

[38] Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. Fine-tuning or retrieval? comparing knowledge injection in llms. *arXiv preprint arXiv:2312.05934*, 2023.

[39] Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *arXiv preprint arXiv:2305.11206*, 2023.

[40] Yizhen Luo, Jiahuan Zhang, Siqi Fan, Kai Yang, Yushuai Wu, Mu Qiao, and Zaiqing Nie. Biomedgpt: Open multimodal generative pre-trained transformer for biomedicine. *arXiv preprint arXiv:2308.09442*, 2023.

[41] Cheng Peng, Xi Yang, Aokun Chen, Kaleb E Smith, Nima PourNejatian, Anthony B Costa, Cheryl Martin, Mona G Flores, Ying Zhang, Tanja Magoc, et al. A study of generative large language model for medical research and healthcare. *arXiv preprint arXiv:2305.13523*, 2023.

[42] Eric Lehman, Evan Hernandez, Diwakar Mahajan, Jonas Wulff, Micah J Smith, Zachary Ziegler, Daniel Nadler, Peter Szolovits, Alistair Johnson, and Emily Alsentzer. Do we still need clinical language models? *arXiv preprint arXiv:2302.08091*, 2023.

[43] Pengyu Cheng, Yifan Yang, Jian Li, Yong Dai, and Nan Du. Adversarial preference optimization. *arXiv preprint arXiv:2311.08045*, 2023.

[44] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*, 2023.

[45] Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, et al. Direct language model alignment from online ai feedback. *arXiv preprint arXiv:2402.04792*, 2024.

[46] Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of rlhf on llm generalisation and diversity. *arXiv preprint arXiv:2310.06452*, 2023.

[47] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[48] meta-llama. Llama recipes. `https://github.com/meta-llama/llama-recipes`. Accessed: Mar, 2024.

[49] Hans W Borchers and Maintainer Hans W Borchers. Package 'pracma'. *Practical numerical math functions, version*, 2(5), 2019.

[50] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

[51] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

[52] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[53] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

[54] Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, and Nathan Lambert. TRL: Transformer Reinforcement Learning. `https://github.com/stas00/ml-engineering`. Accessed: Dec, 2023.

[55] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[56] Kathleen E Walsh, Polina Harik, Kathleen M Mazor, Deborah Perfetto, Milena Anatchkova, Colleen Biggins, Joann Wagner, Pamela J Schoettker, Cassandra Firneno, Robert Klugman, et al. Measuring harm in health care: optimizing adverse event review. *Medical care*, 55(4): 436–441, 2017.

[57] Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020.

[58] Kilem L Gwet. *Handbook of inter-rater reliability: The definitive guide to measuring the extent of agreement among raters*. Advanced Analytics, LLC, 2014.

[59] Kilem L Gwet and Maintainer Kilem L Gwet. Package "irrcac", 2019.

[60] Quoc Duyet Tran, Anil Dolgun, and Haydar Demirhan. The impact of grey zones on the accuracy of agreement measures for ordinal tables. *BMC Medical Research Methodology*, 21: 1–19, 2021.

[61] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

arXiv:2405.00715v1 [cs.CL] 25 Apr 2024

Supplementary Material

Contents

1 Supplementary Method 1: Experiment Chronicles **2**

1.1 Overview . . . 2

1.2 CP . . . 2

1.2.1 CP Experiment 1 . . . 2

1.2.2 CP Experiment 2 . . . 2

1.2.3 CP Experiment 3 . . . 2

1.2.4 CP Experiment 4 . . . 2

1.2.5 CP Experiment 5 . . . 3

1.2.6 CP Experiment 6 . . . 3

1.3 SFT and RLAIF . . . 5

1.3.1 SFT Experiment 1 . . . 5

1.3.2 RLAIF Experiment 1 . . . 5

1.3.3 SFT Experiment 2 . . . 6

1.3.4 RLAIF Experiment 2 . . . 7

1.3.5 RLAIF Experiment 3 . . . 9

1.4 RLHF . . . 12

2 Supplementary Method 2: Prompts to Gemini Pro for Dialogue-G Creation **12**

3 Supplementary Method 3: Prompts to Gemini Pro for Reference Note Generation **13**

4 Supplementary Method 4: Instructions for Collecting Preference Data **14**

5 Supplementary Method 5: Instructions and Scoring Rubric for Physician Reader Study **14**

6 Supplementary Method 6: Instructions for Harm Evaluation **16**

Preprint. Under review.

1 Supplementary Method 1: Experiment Chronicles

1.1 Overview

Our objective is to perform domain- and task-specific adaptation of LLaMA-2 for generating outpatient clinical notes based on patient-physician dialogues. This process is structured around four sequential steps: continued pretraining (CP), supervised finetuning (SFT), reinforcement learning from AI feedback (RLAIF), and reinforcement learning from human feedback (RLHF). Here, we detailed the trial-and-error process throughout our experiments.

1.2 CP

1.2.1 CP Experiment 1

Models: LLaMA-2-13B base and chat models with Low-Rank Adaptation (LoRA).

Dataset: We conducted experiments on two datasets: the complete discharge summary from MIMIC-IV (Discharge-long) and the extracted "Hospital Course" section from discharge summaries (Discharge-short).

Approach: In this experiment, we evaluated the efficacy of pure bf16 training while adhering to the majority of the default hyperparameters specified in the LLaMA-recipes. We used a learning rate (LR) of 3e-4 without a scheduler. Other hyperparameters were selected based on computing resource available at the time of experiments (4 x A6000 or 4 x A100 GPUs), including a global batch size of 16 without gradient accumulation.

Results: We ran into training loss spike on 13B base model that never fully recovered (Supplementary Figure 1 A).

Solution: Restarted pretraining using a mixed precision strategy.

1.2.2 CP Experiment 2

Models and Dataset: Same as experiment 1.

Approach: In this experiment, we implemented mixed precision training, wherein weights and gradients were stored in bf16 format, and optimizer states in fp32. This decision was based on the hypothesis that training with pure bf16 might pose challenges in achieving convergence [1]. The remaining setup parameters were consistent with those outlined in Experiment 1.

Results: During the course of training the 13B-base model, we encountered another spike in training loss, notably in the early stages of the training process (Supplementary Figure 1B). Although the loss quick recovered, it did not revert to its baseline level. For runs that were completed successfully, the outcomes were comparable to those observed in Experiment 1.

Solution: Restarted pretraining using a LR scheduler.

1.2.3 CP Experiment 3

Models and Dataset: Same as experiment 1.

Approach: The standard LLaMA-recipes library does not incorporate a LR scheduler at the time of our work, resulting in a constant LR for each epoch in the previous two experiments. In this iteration, we maintained mixed precision training and introduced a cosine LR scheduler with 200 warm-up steps, followed by a decay to 0% of the peak LR.

Results: All runs were completed successfully (see Figure 3 in the main manuscript), and **we utilized checkpoints from this experiment for SFT and RLAIF**. We observed a significant reduction in training loss when using the Discharge-long dataset, with losses dropping to the 0.9 range. Conversely, for the Discharge-short dataset, the training loss plateaued around 1.4.

1.2.4 CP Experiment 4

Models and Dataset: Same as experiment 1.

Approach: In this experiment, we explored the implementation of pure bf16 training, utilizing the same cosine scheduler as in Experiment 3. Our aim was to ascertain whether this approach could stabilize pure bf16 training, which offers the advantage of reduced VRAM requirements.

Results: A spike in training loss was observed in the 13B-chat model during training with the Discharge-Long dataset (Supplementary Figure 1C).

1.2.5 CP Experiment 5

Models: LLaMA-2-7B base and chat models with full parameter training.

Dataset: Same as experiment 1.

Approach: Motivated by the inferior performance of LoRA compared to full parameter training across various benchmarks (e.g., [2]), we explored the implementation of full parameter training on 7B models, considering their similar VRAM requirements (compared to LoRA training on the 13B models).

Results: In the 7B-chat model runs, we observed a spike in training loss for both mixed precision and pure bf16 training modalities, despite employing a cosine LR scheduler.

Solution: Lowered LR.

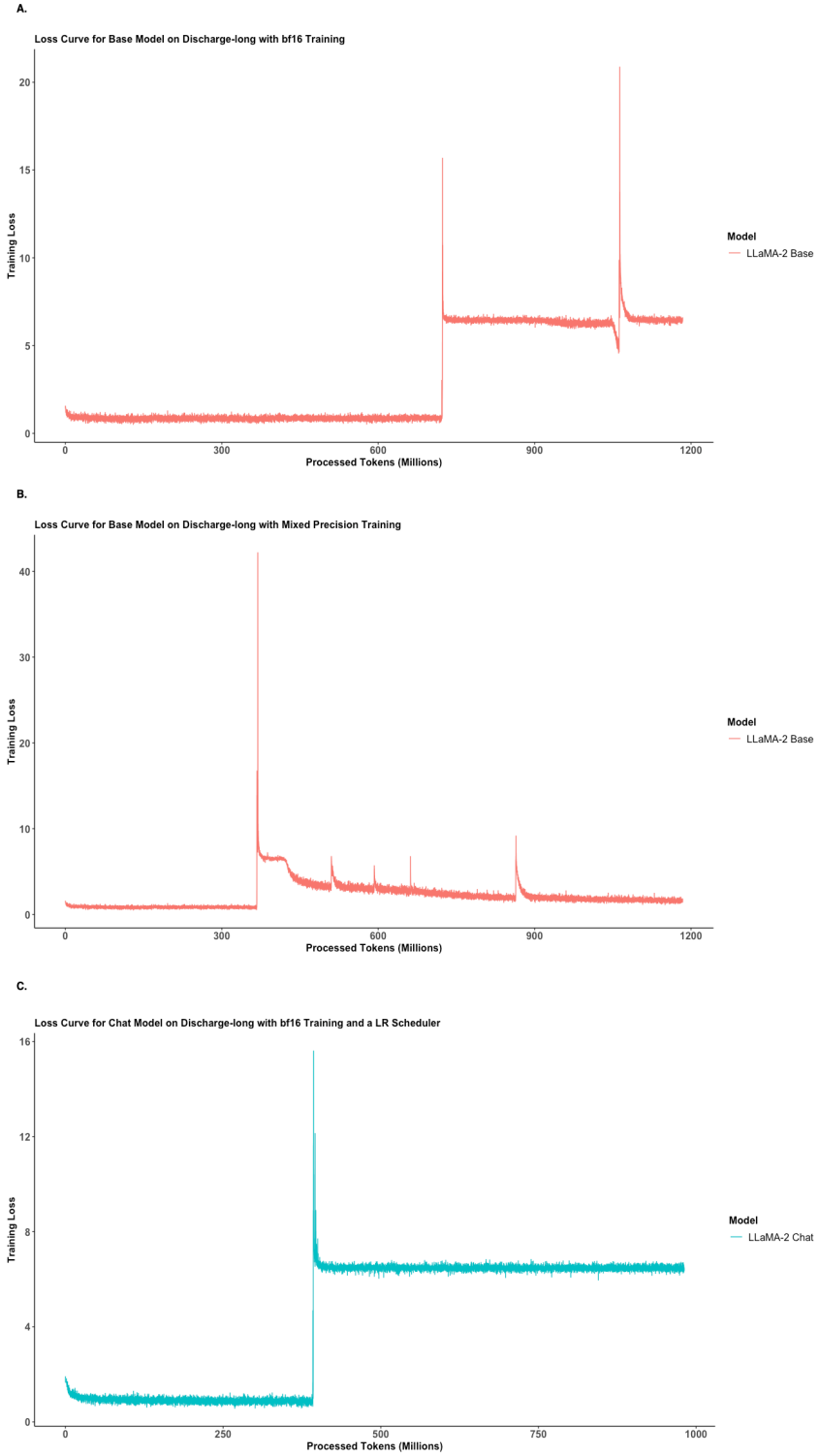
1.2.6 CP Experiment 6

Models: Same as experiment 5.

Dataset: Same as experiment 1.

Approach: We reduced LR to 2e-5, down from 3e-4 in the previous experiment. Of note, the vanilla LLaMA-2 7B model's using a LR of 3e-4 for pretraining.

Results: With the adjusted LR, along with the implementation of pure BF16 training and a cosine LR scheduler, we were able to successfully complete all runs. Checkpoints from these runs were utilized for SFT and RLAIF.



Supplementary Figure 1: Training loss spikes for 13B Models during continued pretraining. A. Loss curve for base model with bf16 training on vanilla LLaMA-recipes without a LR scheduler. B.Loss curve for base model with mixed precision training without a LR scheduler. C. Loss curve for chat model with a LR scheduler and bf16 training. All experiments were performed on Discharge-long dataset. The X-axis represents processed training tokens, and the Y-axis represents training loss. Original loss curves were shown without smoothing.

1.3 SFT and RLAIF

1.3.1 SFT Experiment 1

Models: We utilized the four 13-B checkpoints (base-model/chat-model trained with Discharge-long/Discharge-short) from CP Experiment 3 employing LoRA, and the four 7-B checkpoints from CP Experiment 6 with full parameter training.

Dataset: We combined the training subsets from ACI-BENCH (dialogue n = 67) and Dialogue-G (dialogue n = 1291), then split this data equally for SFT and RLAIF, stratified by data source. Notably, for each dialogue we ask model to generate "Subjective" and "Assessment and Plan" in two separate prompts (therefore two data points per dialogue).

Approach: Instruction tuning was applied to the SFT dataset over 3 epochs, with a LR of 2e-5 for all models, maintaining a fixed LR.

Results: The performance metrics post-SFT are presented in Supplementary Table 1, under the column labeled "R0". Interestingly, the models, after undergoing pretraining and SFT, exhibited inferior performance compared to the out-of-the-box LLaMA-2-chat models. This decline in performance may be attributed to catastrophic forgetting observed during the CP phase.

	Subjective				**Assessment and Plan**			
Models	R0	R1	R2	R3	R0	R1	R2	R3
7B_short	0.2332	0.3373	0.3163	0.2730	0.2259	0.2209	0.3164	0.3410
7B_long	0.2308	0.3619	0.0875	0.3262	0.2030	0.1992	0.2357	0.3205
7B_chat_short	0.2644	0.3471	0.0292	**0.4465**	0.2531	0.2670	0.3370	**0.4436**
7B_chat_long	0.2322	0.4270	0.4072	0.2639	0.2305	0.2841	0.3193	0.3916
13B_short	0.1335	0.3472	0.2353	0.3757	0.1586	0.1873	0.0540	0.3079
13B_long	0.1929	0.3329	0.2234	0.2949	0.1816	0.2602	0.2367	0.2918
13B_chat_short	0.2161	0.3703	0.4168	0.2297	0.2411	0.3901	0.3629	0.3202
13B_chat_long	0.1797	0.3617	0.3342	0.4065	0.2337	0.3381	0.3944	0.3903
Baseline without CP or SFT								
7B	0.0305				0.000			
7B_chat	0.2808				0.3538			
13B	0.0249				0.0049			
13B_chat	0.3114				0.3693			

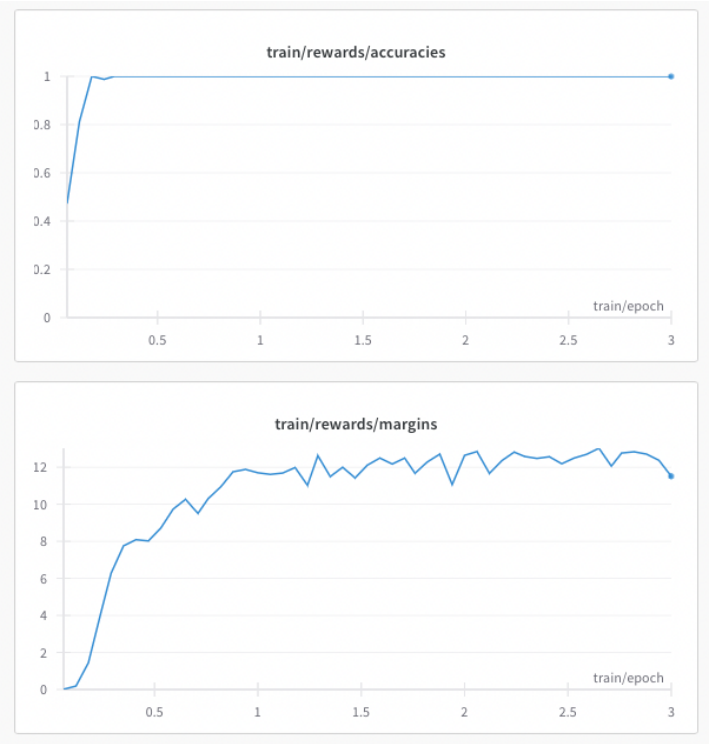
Supplementary Table 1: ROUGE-1 score after RLAIF Experiment 1. Performance reported on validation subset of ACI-BENCH. R0 represent the model after CP and SFT. R1 to R3 represents performance after respective rounds of DistillDirect. Models ended in long were pretrained using Discharge-long dataset, while models ended in short were pretrained using Discharge-short dataset. **Bolded scores** denote the best performance with respect to the task. The experimental setup includes training on the original reference notes with a LR of 2e-5, and 3 epochs per training round. The temperature is set at 1.0 during generation time to calculate ROUGE-1.

1.3.2 RLAIF Experiment 1

Models: We continued with the eight checkpoints derived from SFT Experiment 1. For the 13B models, LoRA was employed, while full parameter training was applied to the 7B models.

Dataset: The RLAIF split of the training dataset as mentioned above.

Approach: RLAIF was conducted using DistillDirect over three rounds. In each round, we first generated notes using our training models, which were designated as "rejected" notes. For ACI-BENCH, the "preferred notes" were the original reference notes in ACI-BENCH. For Dialogue-G, the preference notes were generated by Gemini-pro. For subsequent rounds, we sampled from the newly updated model checkpoint to produce a new set of "rejected" notes, while maintaining the same "preferred" notes. Each round consisted of three epochs, with a LR of 2e-5 in addition to a cosine scheduler.



Supplementary Figure 2: Example train set accuracy and reward margin during DPO with a LR of 2e-5. Examples taken from 13B_chat_short_R3, and all other runs have similar training curve with high accuracy and reward margin early on with this LR.

Results: Performance metrics following RLAIF Experiment 1 are presented in Supplementary Table 1. Although we achieved commendable performance by the third round (notably the best performance came from the 7B-chat_short model), the training process exhibited instability. For instance, the performance in the second round of DistillDirect for the 7B_chat_short model showed a significant decline from the previous round. Across all models, we did not observe a consistent enhancement in performance (as assessed by the ROUGE-1 score on the validation set) with additional rounds of DPO training. Notably, training accuracy reached 100% prematurely, during the first epoch for all models with a high reward margin (as shown in Supplementary Figure 2), indicating potential overfitting.

Solution: Recognizing the instability of the current DistillDirect setup, we considered several potential causes, including issues with data quality/distribution shift, an inappropriate LR leading to overfitting, and reward hacking, among others. Our initial step towards addressing these concerns involved a thorough examination of the training data.

1.3.3 SFT Experiment 2

Models: We restarted SFT using the eight checkpoints after CP as discussed in SFT experiment 1.

Dataset: Upon closely examining the ACI-BENCH and our synthetic dataset, we observed several key points:

1. The quality of reference notes within ACI-BENCH is markedly variable. Certain notes are of poor quality (for example, overly brief), and others adopt syntax or styles not typically found in real clinical notes (for instance, explicitly including a "medical reasoning" section).

2. Additionally, notes from different subspecialties exhibit significant variations in style and format (e.g., a note from internal medicine versus one from orthopedics). While the heterogeneity observed in ACI-BENCH likely mirrors the diversity encountered in clinical practice (where different physicians may write notes very differently), it poses a challenge

to use these notes as a "gold standard" for training a language model to replicate such documentation.

3. Similar issues were also noted in our augmented Dialogue-G dataset created using Gemini-pro. Despite employing prompts with clear instructions regarding note format, minor inconsistencies still emerged.

Approach: Through discussions with our physician co-authors, we concluded that before AI can assist physicians in generating notes, the physicians themselves must determine what constitutes a "best practice" note format. It's acknowledged that the "best practice" might diverge from existing notes for valid reasons (notably, recognizing that physicians can also produce suboptimal notes). With this perspective, we revised our prompts and provided a single-shot example to Gemini-pro to foster "constrained" note generation tailored towards a specific style. This new note style was collectively endorsed by our physician authors as superior to many examples within ACI-BENCH. Consequently, we directed Gemini-pro to generate new reference notes from the dialogues in ACI-BENCH, discontinuing the use of its original notes. At this stage, we also regenerated reference notes for Dialogue-G using Gemini Pro. Initially, we considered having our physicians edit Gemini-pro's notes before employing them for training. However, we abandoned this idea due to the time-intensive nature of the task and the satisfactorily high quality of outputs from Gemini-pro with our revised prompt.

Results: Utilizing the newly formulated dataset, we performed SFT for 3 epochs, adhering to the same parameters as before. The performance metrics post-SFT are detailed in Supplementary Table 2, under the column labeled "R0".

	Subjective				**Assessment and Plan**			
Models	R0	R1	R2	R3	R0	R1	R2	R3
7B_short	0.1998	0.4894	0.5198	0.3792	0.2617	0.4885	0.0917	0.4100
7B_long	0.1949	0.4688	0.0594	0.3920	0.2446	0.3705	0.3053	0.4040
7B_chat_short	0.2961	0.4816	0.4946	0.0586	0.3157	0.5201	0.4763	0.3867
7B_chat_long	0.3067	0.5051	0.0964	0.0773	0.2988	0.4907	0.4275	0.3628
13B_short	0.2463	0.5064	0.4904	0.4066	0.2277	0.5252	0.4059	0.4493
13B_long	0.2525	0.4199	0.5113	0.3309	0.2565	0.4840	0.4610	0.3566
13B_chat_short	0.3475	0.5115	0.5030	**0.5441**	0.3055	0.5277	0.4422	**0.5422**
13B_chat_long	0.3463	0.4856	0.5308	0.3863	0.3620	0.4752	0.4730	0.4633
Out-of-box models with SFT only without CP								
13B	0.2813	0.5303	0.3967	0.5108	0.2977	0.5265	0.3510	0.5060
13B_chat	0.2874	0.5073	0.5287	**0.5452**	0.3301	0.5069	0.4895	**0.5597**
Out-of-box models without SFT or RLHF								
13B	0.0329				0.0100			
13B_chat	0.3585				0.4543			

Supplementary Table 2: ROUGE-1 score after RLAIF Experiment 2. Performance reported on validation subset of ACI-BENCH. R0 represent the model after CP and SFT. R1 to R3 represents performance after respective rounds of DistillDirect. Models ended in long were pretrained using Discharge-long dataset, while models ended in short were pretrained using Discharge-short dataset. **Bolded scores** denote the best performance with respect to the task. The experimental setup includes training on the new reference notes with a LR of 2e-5, and 3 epochs per training round. The temperature is set at 1.0 during generation time to calculate ROUGE-1.

1.3.4 RLAIF Experiment 2

Models: The eight checkpoints from SFT experiment 2. We used LoRA for 13B models and full parameter training for 7B models.

Dataset: New dataset as described in SFT experiment 2.

Approach: Similar to RLAIF Experiment 1, but this iteration utilized the newly generated notes by Gemini-pro as "preferred" notes. Additionally, to mitigate concerns of overfitting, we limited the training to 1 epoch for each round, as opposed to the 3 epochs per round in RLAIF Experiment 1.

Results: A number of intriguing observations can be made from this iteration (Supplementary Table 2).

- **7B vs 13B models:** With the current LR, the 7B models exhibited significant instability during DistillDirect. For instance, the 7B_short model demonstrated a notable performance degradation from round 1 to round 2. Upon manual evaluation of its outputs, the model incorrectly generated "Subjective" sections when tasked with producing "Assessment and Plan" and frequently returned null outputs. Given the more stable training observed with 13B models utilizing LoRA, along with their superior performance and reduced storage requirements compared to full parameter training for 7B models, **we have decided to proceed exclusively with 13B models henceforth**.
- **Best Performer by ROUGE-1 :** Among our CP models, the 13B_chat model pre-trained with the Discharge-short dataset emerged as the top performer. However, the overall best model was the 13B_chat model following SFT and three rounds of DistillDirect, **but without CP**. It achieved ROUGE-1 scores of 0.5452 for "Subjective" and 0.5597 for "Assessment and Plan". Naturally, this led us to question the value of CP in our specific task. Such skepticism seemed justified, particularly given the differences between our training corpus of discharge summaries and that of outpatient clinical notes, which could be deemed out-of-distribution for our task. Interestingly, a more thorough review and manual analysis of the notes by our physician author unveiled intriguing subtleties.
- **Possible Best Performer by Physician Review:** The quality of the notes generated by the top-performing models (e.g., 13B_chat_R3, 13B_short_R1, 13B_chat_short_R3, 13B_chat_long_R2) appears nearly indistinguishable to our physician author. Overall, we find the "Subjective" sections of these notes to be satisfactory and potentially suitable for clinical application. However, the primary deficiency in achieving clinical-ready notes lies within the "Assessment and Plan" sections, particularly regarding medical reasoning and certain linguistic nuances. Although somewhat subjective, we felt the outputs from 13B_short_r1 bear the greatest resemblance to an actual provider's note. **This finding underscores the limitations of quantitative metrics in tasks involving domain-specific language generation**.
- **Persistent DistillDirect Training Instability:** Despite the creation of a more uniform dataset, DistillDirect training remained susceptible to fluctuations. This instability was more pronounced for 7-billion parameter models but was also observed in 13-billion parameter models (Supplementary Table 2). The 13B_long model serves as an example, where the ROGUE-1 score dropped from 0.5113 in round 2 to 0.3309 in round 3 for the "Subjective" section. Reducing the number of epochs per DistillDirect round from three to one did not mitigate this issue. In all DistillDirect rounds, training set accuracy rapidly reached a perfect score (1.0) halfway through the first epoch, accompanied by a high reward margin (as illustrated in the initial third of the curve shown in Supplementary Figure 2). This observation again suggests potential overfitting. Interestingly, similar rapid accuracy increases were reported during Zephyr training, and overfitting did not negatively impact performance on downstream tasks [3].
- **Reflection on LR Selection:** The optimal LR for DPO has yet to be established in literature, with only a limited number of LLMs having undergone DPO training to date. Notably, given the small size of our training set, our initial LR choice of 2e-5 is larger than those selected in Zephyr and Tulu-2, where both models opted for an LR of 5e-7 [4, 3]. The Tulu-2 paper mentioned that a "slow LR... is required for stable and effective DPO training." Conversely, in another study, Gaudi-2 employed a significantly higher LR (5e-4, compared to an LR of 1e-4 during its SFT), despite using the same training data as Zephyr, and still achieved a favorable response [5].
- **Generative Parameter Matters:** We investigated the influence of generative parameters on model performance. Specifically, we experimented with a lower temperature setting of 0.6 compared to the baseline of 1.0 (Supplementary Table 3), while maintaining other parameters the same (multinomial sampling with top-k = 50, top-p = 1.0, and repetition penalty = 1.2). We refrained from employing fast kernels such as flash attention due to observed declines in output quality. Lowering the temperature resulted in a consistent increase in ROUGE-1 scores across all models. This finding suggests that optimizing

generation-related hyperparameters through a dedicated search process has the potential to further enhance performance.

Solution: Ideally, at this stage, our goal would be to select the best model, sample its generations under various settings, and forward these samples to physicians for labeling in preparation for the RLHF. However, the absence of a stable DistillDirect setup presents a significant challenge. Moreover, there is no assurance that further DPO training with real human preference data will result in performance improvement. Additionally, we have yet to confirm that we have identified the optimal model using the existing training dataset. Therefore, we have decided to conduct a limited LR search specifically for DistillDirect.

	Subjective				**Assessment and Plan**			
Models	R0	R1	R2	R3	R0	R1	R2	R3
13B_short	0.4006	0.5423	0.5186	0.4702	0.4032	0.5683	0.4039	0.5242
13B_long	0.3570	0.4799	**0.5779**	0.3981	0.4298	0.5266	0.4688	0.3780
13B_chat_short	0.4390	0.5455	0.5248	0.5560	0.4825	0.5565	0.5236	**0.5694**
13B_chat_long	0.4272	0.5149	0.5423	0.4188	0.4459	0.5487	0.5322	0.4852
Out-of-box models with SFT only without CP								
13B	0.3671	0.5469	0.4216	0.5508	0.4335	0.5804	0.3833	0.4913
13B_chat	0.4077	0.5351	**0.5582**	0.5466	0.4383	0.5522	0.4907	**0.5847**

Supplementary Table 3: ROUGE-1 score after RLAIF Experiment 2 with lower temperature. Performance reported on validation subset of ACI-BENCH. R0 represent the model after CP and SFT. R1 to R3 represents performance after respective rounds of DistillDirect. Models ended in long were pretrained using Discharge-long dataset, while models ended in short were pretrained using Discharge-short dataset. **Bolded scores** denote the best performance with respect to the task. The experimental setup includes training on the new reference notes with a LR of 2e-5, and 1 epoch per training round. The temperature is set at 0.6 during generation time to calculate ROUGE-1, using the same model checkpoints as in Supplementary Table 2
.

1.3.5 RLAIF Experiment 3

Models: 13B model checkpoints from SFT experiment 2.

Dataset: New dataset as described in SFT experiment 2.

Approach: At this stage, our efforts were concentrated on a limited LR search for DistillDirect. We evaluated an LR of 5e-6 and 5e-7 and compared the effects of 1 epoch versus 3 epochs in each round across three rounds. After each round of training, we resampled outputs from the updated model checkpoint to serve as "rejected" samples for the subsequent round of DPO, while continuing to use notes produced by Gemini-pro as the "preferred" samples.

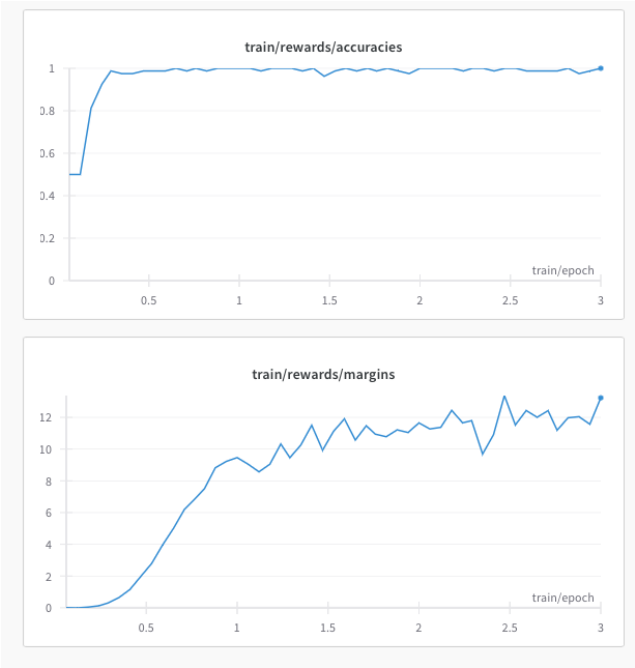
Results:

- **LR of 5e-7:** Training set accuracy reached 0.98 by the end of the second epoch. After three epochs, the reward margin only attained a level of 0.25 (Supplementary Figure 3). This very small, albeit popular, LR was found to be insufficient in our experimental setup, yielding very slow performance improvements regardless of whether we conducted 1 epoch per round or 3 epochs per round of training (Supplementary Table 4).

Supplementary Figure 3: Example train set accuracy and reward margin during DPO with a LR of 5e-7. Examples taken from 13B_chat_short_R1.Training set accuracy reaches 0.98 at the end of 2nd epoch. After 3 epochs, the reward margin only reaches a level of 0.25.

- **LR of 5e-6:** Training set accuracy reached 0.95 midway through the second epoch. After three epochs, the reward margin achieved a level of 12 (refer to Supplementary Figure 4). Although the training curve bears resemblance to that associated with a LR of 2e-5 (see Supplementary Figure 4 abd Supplementary Figure 2), this lower LR demonstrated greater stability. This was particularly evident in the experiments conducted with 1 epoch per round of training, where performance across all models either exhibited steady improvement or achieved convergence, devoid of the abrupt declines observed with an LR of 2e-5 (Supplementary Table 4). Notably, several models reached their performance peak in round 2, indicating that 2 to 3 rounds of training might be an optimal range for our experiment setup.
- **Impact of Temperature:** Consistent with previous experiments, model performance improves when a lower temperature was applied during generation (see Supplementary Table 5). This observation motivated us to use lower temperature for RLHF generations.

Solution: We identified the LR of 5e-6 with a 1-epoch per round as the stable setup for DistillDirect in our experiment. Upon manual examination of outputs from all model checkpoints within this configuration, **we selected the 13B_chat_short model from round 3 for the next stage of RLHF** (Supplementary Table 4).



Supplementary Figure 4: Example train set accuracy and reward margin during DPO with a LR of 5e-6. Examples taken from 13B_chat_short_R1.Training set accuracy reaches 0.98 at the end of 2nd epoch. After 3 epochs, the reward margin only reaches a level of 0.25.

	Subjective				Assessment and Plan			
Models	R0	R1	R2	R3	R0	R1	R2	R3
Lr: 5e-6, 3 epochs per round								
13B_short	0.2463	0.5124	0.4820	0.5310	0.2277	0.5123	0.4547	0.4808
13B_long	0.2525	0.4406	**0.5423**	0.2974	0.2565	0.4652	0.4984	0.3316
13B_chat_short	0.3475	0.4633	0.4727	0.5422	0.3055	**0.5293**	0.5012	0.5086
13B_chat_long	0.3463	0.4936	0.5097	0.5327	0.3620	0.5290	0.5008	0.5247
Lr: 5e-6, 1 epoch per round								
13B_short	0.2463	0.3708	0.4794	0.4775	0.2277	0.4318	0.4995	0.4956
13B_long	0.2525	0.3730	0.4427	0.4494	0.2565	0.3609	0.4750	0.4578
13B_chat_short	0.3475	0.4254	0.4804	**0.4878**	0.3055	0.4329	0.5140	**0.5182**
13B_chat_long	0.3463	0.4135	0.4600	0.4601	0.3620	0.4642	0.4843	0.4662
Lr: 5e-7, 3 epochs per round								
13B_short	0.2463	0.2546	0.2655	0.2744	0.2277	0.2618	0.2573	0.3002
13B_long	0.2525	0.2151	0.2382	0.2115	0.2565	0.2458	0.3149	0.2755
13B_chat_short	0.3475	0.3518	**0.3718**	0.3428	0.3055	0.3398	0.3449	**0.3840**
13B_chat_long	0.3463	0.3311	0.2984	0.3296	0.3620	0.3310	0.3719	0.3747
Lr: 5e-7, 1 epoch per round								
13B_short	0.2463	0.2864	0.2921	0.2644	0.2277	0.2599	0.2731	0.2735
13B_long	0.2525	0.2600	0.2333	0.1839	0.2565	0.2473	0.2216	0.2453
13B_chat_short	**0.3475**	0.3117	0.3388	0.3356	0.3055	0.3303	0.3370	0.3231
13B_chat_long	0.3463	0.3090	0.2943	0.3134	0.3620	0.3174	0.3194	**0.3693**

Supplementary Table 4: ROUGE-1 score after RLAIF Experiment 3. Performance reported on validation subset of ACI-BENCH. R0 represent the model after CP and SFT. DistillDirect R1 to R3 represents performance after respective rounds of DistillDirect. Models ended in long were pretrained using Discharge-long dataset, while models ended in short were pretrained using Discharge-short dataset. **Bolded scores** denote the best performance with respect to the task. The experimental setup includes training on the new reference notes with a variable learning rates and training epochs. The temperature is set at 1.0 during generation time to calculate ROUGE-1.

1.4 RLHF

The model checkpoints selected for RLHF have undergone training through a sequence of stages: CP Experiment 3, followed by SFT Experiment 2, and then RLAIF Experiment 3. During the RLHF phase, we experimented with different temperature settings and observed that physicians tend to prefer outputs generated at lower temperatures. Consequently, we decided to adopt more deterministic generation parameters for the physician reader study.

	Subjective				**Assessment and Plan**			
Models	R0	R1	R2	R3	R0	R1	R2	R3
Lr: 5e-6, 3 epochs per round								
13B_short	0.4006	0.5165	0.5143	0.5349	0.4032	0.5471	0.4996	0.5246
13B_long	0.3570	0.5049	0.5679	0.4289	0.4298	0.5064	0.5102	0.3976
13B_chat_short	0.4390	0.5343	0.5309	0.5614	0.4825	0.5524	0.5266	**0.5575**
13B_chat_long	0.4272	0.4815	**0.5759**	0.5605	0.4459	0.5299	0.5510	0.5294
Lr: 5e-6, 1 epoch per round								
13B_short	0.4006	0.4971	0.5145	0.5200	0.4032	0.4950	0.5464	0.5222
13B_long	0.3570	0.4449	0.5126	0.5000	0.4298	0.5123	0.5190	0.5285
13B_chat_short	0.4390	0.5155	0.5167	**0.5352**	0.4825	0.5380	0.5392	**0.5411**
13B_chat_long	0.4272	0.4863	0.4980	0.5077	0.4459	0.5094	0.5255	0.5346
Lr: 5e-7, 3 epochs per round								
13B_short	0.4006	0.3760	0.3771	**0.4573**	0.4032	0.4390	0.4281	0.4315
13B_long	0.3570	0.4022	0.3909	0.3693	0.4298	0.4154	0.4428	0.4411
13B_chat_short	0.4390	0.4434	0.4482	0.4394	0.4825	0.4761	0.4625	**0.4833**
13B_chat_long	0.4272	0.4398	0.4058	0.4282	0.4459	0.4763	0.4633	0.4545
Lr: 5e-7, 1 epoch per round								
13B_short	0.4006	0.4241	0.4112	0.4140	0.4032	0.4345	0.4172	0.4107
13B_long	0.3570	0.4099	0.4175	0.4039	0.4298	0.3671	0.3865	0.4285
13B_chat_short	0.4390	0.4291	**0.4541**	0.4337	0.4825	0.4569	**0.4826**	0.4450
13B_chat_long	0.4272	0.4188	0.4048	0.3986	0.4459	0.4606	0.4552	0.4446

Supplementary Table 5: ROUGE-1 score after RLAIF Experiment 3 with a lower temperature. Performance reported on validation subset of ACI-BENCH. R0 represent the model after CP and SFT. R1 to R3 represents performance after respective rounds of DistillDirect. Models ended in long were pretrained using Discharge-long dataset, while models ended in short were pretrained using Discharge-short dataset. **Bolded scores** denote the best performance with respect to the task. The experimental setup includes training on the new reference notes with a variable learning rates and training epochs. The temperature is set at 0.6 during generation time to calculate ROUGE-1 using the same model checkpoints as in Supplementary Table 4.

2 Supplementary Method 2: Prompts to Gemini Pro for Dialogue-G Creation

We presented the prompt sent to Gemini Pro for generating dialogues of Dialogue-G in Supplementary Table 6. We subsequently used these dialogues to create reference notes of Dialogue-G using prompts in Supplementary Table 7.

Category	**Prompt**
Dialogue	Generate a synthetic patient-physician clinical dialogue encounter based on the clinical note below. Make sure all pertinent details are represented in the generated dialogue, so that a physician can easily write up the provided note. Pay close attention to make sure all details in the 'ASSESSMENT AND PLAN', 'HISTORY OF PRESENT ILLNESS' and 'SUBJECTIVE' parts of the note are reflected in the dialogue. You may expand details to make the dialogue resemble a real clinical encounter. Denote doctor as [doctor] and patient as [patient].

Supplementary Table 6: Prompts to Gemini Pro for Dialogue-G Creation.

3 Supplementary Method 3: Prompts to Gemini Pro for Reference Note Generation

We presented the prompt used by Gemini Pro to create reference notes in Supplementary Table 7.

Category	Prompt
Subjective	You are a physician writing a clinical note based on a dialogue with the patient. Only write the "SUBJECTIVE" part of note, which include the section of [CHIEF COMPLAINT] and [HISTORY OF PRESENT ILLNESS]. Only include information contained in the dialogue. Follow the format as the example below: SUBJECTIVE CHIEF COMPLAINT Annual health maintenance examination. HISTORY OF PRESENT ILLNESS The patient is a pleasant [age]-year-old male who presents for his annual health maintenance examination. He reports no new complaints today. He denies any recent changes in his hearing. He continues to take niacin for his dyslipidemia, and he has had no problems with hemorrhoids in the last 6 months. He also denies any problems with concha bullosa of the left nostril or septal deviation.
Assessment and Plan	You are a physician writing a clinical note based on a dialogue with the patient. Only write the "ASSESSMENT AND PLAN" section of note. List each medical problem separately. Under each problem, include assessment (such as medical reasoning) and plan (both diagnostic and therapeutic). At the end, may include a short section on follow up instruction when applicable. Only include information contained in the dialogue. Follow the format as the example below: ASSESSMENT AND PLAN: 1. Possible COPD exacerbation Assessment: Increased work of breathing with wheezing on exam, suggesting COPD exacerbation. He does have frequent COPD exacerbation in the past. Differential diagnosis include pneumonia (though no fever or cough), PE (though no risk factors) or simple viral infection. Plan: - WIll obtain CXR. - Will start duoneb therapy and oral prednisone 30mg Qday. 2. Hypertension Assessment: The patient's blood pressure is well controlled. Plan: - Continue lisinopril 20mg Qday. Follow-up instructions: - return to clinic in 1 week, or sooner of failed to response with current treatment.

Supplementary Table 7: Prompts to Gemini Pro for Reference Note Generation.

4 Supplementary Method 4: Instructions for Collecting Preference Data

We presented the instructions used for collecting physician preference data for RLHF in Supplementary Table 8.

Instructions
1. Please first read the dialogue and then pick your most and least preferred notes. Most conversation occurred in the outpatient setting. 2. We will only look at the "Subjective" and "Assessment and Plan" parts of a note. There will be a separate row for "Subjective" and "Assessment and Plan", respectively. 3. On each row, you will be given three notes generated by LLMs. Pick the MOST preferred note, and the LEAST preferred note by selecting the corresponding note number in the columns of "Preferred" and "Rejected". 4. Focus on whether the clinical note accurately reflected information from the conversation. Ignore any error related to medical knowledge, as long as the information was mentioned in the conversation. 5. Base your preference on factors like clinical readiness, correctness, and adherence to the desired format, including: a. Clinical readiness: Is the note ready for clinical use and does it capture important information? b. Correctness: Does the note include less false information? c. Adherence to format: Does the "Subjective" section include "Chief complaints" and "History of present illness"? Does the "Assessment and Plan" section list each problem separately and include "Assessment:" and "Plan:" with the required details? 6. Make your preference judgement from a clinician's perspective, considering which note would be most/least helpful to you.

Supplementary Table 8: Clinician Note Preference Instruction.

5 Supplementary Method 5: Instructions and Scoring Rubric for Physician Reader Study

We presented the instructions used for the final physician reader study in Supplementary Table 9.

Instruction and rubrics

1. In each row you will be given a synthetic outpatient patient-provider dialogue from ACI-BENCH, and three clinical notes based on the same dialogue. Two notes are generated by large language model, and one note is written by real physician. We have performed randomization of the notes (so that notes from the column of note_1 are from different sources) and simple processing to unify the format of notes.
2. The dialogues from ACI-BENCH include conversations with (a) calls to a virtual assistant (b) unconstrained directions or discussions with a scribe, and (c) natural conversations between a doctor and patient.
3. There are 40 dialogues. For each dialogue, we will evaluate "Subjective" and "Assessment and Plan" parts of the note in separate rows. Therefore, there are 80 rows in total.
4. For each row, you will first read the entire dialogue and then read the 3 notes. You will subsequently score the quality of each note for the 3 axes of "accuracy", "completeness" and "readiness for real-world use". For each axis, you will give a score of 1 to 5 (very poor, poor, acceptable, good, or very good), where higher number suggests better quality. It is OK to give the same score for different notes if you feel they are of similar quality. For each row since there are 3 notes, you will give total 9 scores. In each row there is a section of "Comment" for you to free text any feedback if you feel like to.
5. **Accuracy**: For this axis, answer the question "does the factual information from clinical note accurately match that from the dialogue?" A note is accurate if it doesn't say things that aren't in the dialogue, it doesn't mix up facts, and generally is not misleading. It might be acceptable if the note contains reasonable medical reasoning in the section of "Assessment", for example in describing differential diagnosis. Please ignore any medical knowledge error, as long as the information was discussed in the dialogue.

   Rubric:
   - Score of 1 (very poor): The note contains a significant amount of content that is either factually incorrect, fabricated, or disconnected from the dialogue.
   - Score of 3 (acceptable): The note contains some minor content that is either factually incorrect, fabricated, or disconnected from the dialogue.
   - Score of 5 (very good): The note has no incorrect statements or misleading implications.
6. **Completeness**: For this axis, answer the question "how well does the note cover the important information from the dialogue?" An ideal clinical note would contain all clinically important information represented in the dialogue. Also, just as in real-world scenario, a good clinical note could be short but pertinent. A note has poor coverage if someone reading only the note would be missing several important pieces of information about the clinical encounter. Give your score based on what is typically expected from a clinical note.

   Rubric:
   - Score of 1 (very poor): The note is missing a significant amount of important clinical information from the dialogue.
   - Score of 3 (acceptable): The note is missing some minor piece of clinical information from the dialogue.
   - Score of 5 (very good): The note covers all important clinical information from the dialogue, as you would expect from a real-world note.
7. **Readiness for real-world use**: For this axis, answer the question "which note is most ready for clinical use in the real-world scenario?" Answer this question imagine you are adopting AI-generated clinical notes for your everyday clinical work, and you will proofread and make edits to these notes before file into medical record. In this workflow, which note would you prefer the most? For example, this might be the note that meet your style and carries the most pertinent information without note bloating. Or a best note for you might be the one that requires the least amounts of edits from you (even if it contains some minor error). In other words, you can think of this as scoring the overall quality of the note for the workflow.

   Rubric:
   - Score of 1(very poor): The note is impossible to use or would require significant edits from you.
   - Score of 3 (acceptable): The note requires some edits from you.
   - Score of 5 (very good): The note is ready for clinical use without any further edits from you.

Supplementary Table 9: Instructions for physician reader study.

6 Supplementary Method 6: Instructions for Harm Evaluation

We presented the instructions used for the harm evaluation in Supplementary Table 10.

Instructions
For those notes that you gave a score of less than 5 for "Accuracy" or "Completeness", please answer two questions: Suppose the note is used in the standard clinical workflow, what would be: 1. **"... extent of possible harm?"** **Rubric**: • Score of 1: None. • Score of 2: Mild or moderate harm. • Score of 3: Severe harm or death. 2. **"... likelihood of possible harm?"** **Rubric**: • Score of 1: Low. • Score of 2: Medium. • Score of 3: High.

Supplementary Table 10: Harm Evaluation Instruction.

References

[1] Pedram Zamirai, Jian Zhang, Christopher R Aberger, and Christopher De Sa. Revisiting bffloat16 training. 2020.

[2] Tianyu Han, Lisa C Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K Bressem. Medalpaca–an open-source collection of medical conversational ai models and training data. *arXiv preprint arXiv:2304.08247*, 2023.

[3] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*, 2023.

[4] Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A Smith, Iz Beltagy, et al. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *arXiv preprint arXiv:2311.10702*, 2023.

[5] Intel Analytics Software. The practice of supervised finetuning and direct preference optimization on habana gaudi2. `https://medium.com/intel-analytics-software`, 2023. Accessed: Feb, 2024.